

ANALISIS DAN PERANCANGAN APLIKASI OPNAME AUDIT ASET NON-UNIT BERBASIS MOBILE APPS

Yulmansyah Rizky Rahman¹, Lutfi Harris²
Universitas Brawijaya, Malang, Indonesia
Email: yulmansyah.rizky@gmail.com

Keywords

AppSheet, SDLC, internal audit, asset audit, low-code development, process optimization.

AppSheet, SDLC, internal audit, asset audit, low-code development, process optimization.

Abstrak

This paper aims to analyze and design a new application to improve the non-unit asset opname audit procedure within PT XYZ's Internal Audit Division. The application, developed using AppSheet (a low-code development platform), streamlines the traditional audit method, which involves multiple time-consuming steps. By employing the SDLC (classical waterfall) methodology, the new application simplifies and automates various stages of the process, resulting in a more efficient and user-friendly workflow. The application design enhances the workflow of the audit procedure by eliminating steps that does not have significant added value.

Penelitian ini bertujuan untuk menganalisis dan merancang sebuah aplikasi baru untuk mendukung prosedur opname audit aset non-unit di Divisi Audit Internal PT XYZ. Aplikasi ini dikembangkan menggunakan AppSheet, sebuah platform pengembangan low-code, dan bertujuan untuk menyederhanakan metode audit tradisional yang melibatkan beberapa langkah yang memakan waktu. Dengan menerapkan metodologi SDLC (classical waterfall), aplikasi baru ini menyederhanakan dan mengotomatiskan berbagai tahapan proses, menghasilkan alur kerja yang lebih efisien dan ramah pengguna. Rancangan aplikasi yang dibuat dapat meningkatkan alur kinerja prosedur audit melalui eliminasi terhadap langkah-langkah yang tidak memiliki nilai tambah signifikan.

1. PENDAHULUAN

Seiring perkembangan zaman, perusahaan di berbagai sektor terus berinovasi untuk meningkatkan performa, pangsa pasar, dan keunggulan kompetitif. Penelitian menunjukkan bahwa inovasi berdampak positif pada produktivitas dan pendapatan negara maju maupun berkembang. Salah satu inovasi yang semakin banyak digunakan adalah aplikasi bisnis untuk meningkatkan efisiensi prosedur operasional. Tidak hanya perusahaan besar, bisnis kecil juga telah memanfaatkan aplikasi untuk keperluan seperti penjadwalan barang, pemantauan arus kas, dan pengelolaan persediaan.

Pengembangan aplikasi untuk mendukung aktivitas bisnis tertentu menunjukkan tingkat inovasi yang dicapai oleh perusahaan. Aplikasi dapat meningkatkan efisiensi, ketepatan, dan fleksibilitas dalam menghadapi tantangan bisnis. Penggunaan teknologi informasi juga memungkinkan pengumpulan data yang lebih reliabel dan mengurangi kesalahan dibandingkan dengan proses manual. Salah satu contoh implementasi aplikasi dapat ditemukan di PT XYZ, perusahaan penyewaan kendaraan korporasi yang mengelola lebih dari 28.000 kendaraan dan 4.000 pengemudi di berbagai kota besar Indonesia. PT XYZ memiliki Departemen Audit Internal yang bertanggung jawab atas audit operasional dan kepatuhan.

Salah satu alur kerja utama dalam departemen ini adalah opname audit aset non-unit, yaitu pemeriksaan kondisi aset berwujud yang dicatat dalam berkas fisik sebelum dianalisis lebih lanjut menggunakan komputer. Proses ini dinilai memakan waktu dan mengandung banyak langkah redundan, terutama karena dilakukan hingga 18 kali dalam setahun. Aset berwujud seperti mesin, komputer, dan kendaraan sangat penting bagi operasional perusahaan sehingga pengelolaannya memerlukan pengendalian internal yang baik. Namun, metode tradisional dengan checklist berbasis kertas memiliki banyak kelemahan, seperti risiko human error, minimnya audit trail, dan keterlambatan proses input data ke sistem.

Sistem berbasis kertas juga menyebabkan pemborosan sumber daya dan menghambat efisiensi audit internal. Kesalahan pencatatan dapat mengurangi reliabilitas data, yang berakibat pada pengambilan keputusan yang kurang akurat. Selain itu, tidak adanya sistem kredensial dalam checklist berbasis kertas menyebabkan kurangnya akuntabilitas dan meningkatkan risiko ketidakpatuhan. Penundaan dalam penginputan data ke sistem juga dapat menghambat pemeliharaan aset, menyebabkan malfungsi, atau keterlambatan penggantian aset yang sudah tidak layak pakai.

Untuk mengatasi permasalahan tersebut, solusi berbasis cloud diusulkan sebagai alternatif. Sistem cloud-based memiliki keunggulan dalam hal keamanan dan skalabilitas dibandingkan dengan metode tradisional. Keamanan sistem tradisional bergantung pada kejujuran karyawan dan efektivitas kontrol internal, sedangkan sistem cloud-based memiliki perlindungan data yang lebih baik dan memungkinkan skalabilitas tanpa investasi besar dalam infrastruktur. Meskipun demikian, ada beberapa tantangan dalam mengadopsi sistem cloud, seperti keterbatasan

keterampilan karyawan dalam menggunakan teknologi baru, ketergantungan pada koneksi internet yang stabil, serta kesulitan dalam migrasi data dari sistem lama.

Fungsi audit internal yang bersifat labor-intensive menghadapi tantangan dalam menjaga efisiensi dan akurasi kerja. Oleh karena itu, penggunaan aplikasi yang mengotomatisasi langkah-langkah repetitif dan meningkatkan kemudahan analisis data dapat meringankan beban auditor. Solusi ini memungkinkan auditor untuk lebih fokus pada aktivitas strategis seperti penilaian risiko dan komunikasi dengan auditan.

Dalam konteks ini, Low-code Application Development Platforms (LCDPs) menjadi pilihan yang tepat karena memungkinkan pembuatan aplikasi tanpa memerlukan keterampilan pemrograman yang mendalam. Salah satu platform yang dipilih adalah AppSheet, yang memberikan fleksibilitas dalam modifikasi aplikasi serta integrasi dengan berbagai layanan seperti Google Spreadsheets dan SQL Database. AppSheet dipilih karena beberapa keunggulan utama, termasuk kemampuannya untuk menyimpan data secara real-time, fleksibilitas dalam modifikasi aplikasi tanpa memerlukan keterampilan pemrograman, serta adanya sistem kredensial untuk meningkatkan akuntabilitas dan menyediakan audit trail yang memadai.

Keunggulan lain dari AppSheet adalah pengurangan risiko kehilangan data akibat penggunaan kertas serta efisiensi waktu dalam persiapan prosedur audit. Dengan menggunakan aplikasi ini, checklist tidak lagi perlu dicetak dalam bentuk fisik, sehingga auditor dapat langsung mengakses dan menginput data melalui perangkat digital. Dengan demikian, proses opname audit aset non-unit dapat berjalan lebih cepat dan efisien.

Berdasarkan permasalahan yang telah diuraikan, penelitian ini bertujuan untuk menyediakan alternatif berbasis teknologi guna meningkatkan efektivitas dan efisiensi dalam audit aset non-unit. Implementasi aplikasi audit berbasis mobile apps diharapkan dapat mengatasi masalah redundansi dalam proses audit tradisional dan memberikan solusi yang lebih praktis bagi perusahaan. Dengan adanya inovasi ini, perusahaan dapat meningkatkan akurasi pencatatan aset, mempercepat pengambilan keputusan, serta mengurangi risiko kesalahan dalam proses audit.

Rumusan masalah dalam penelitian ini meliputi: (1) apakah aplikasi penunjang prosedur opname audit aset non-unit dapat menyelesaikan masalah redundansi dalam audit tradisional, dan (2) bagaimana cara merancang aplikasi yang efektif untuk mendukung prosedur opname audit aset non-unit. Adapun tujuan penelitian ini adalah

untuk mengetahui efektivitas aplikasi dalam mengatasi masalah audit tradisional serta merancang aplikasi yang dapat mendukung prosedur opname audit aset non-unit secara optimal.

2. METODE PENELITIAN

PT XYZ adalah perusahaan transportasi terbesar di Indonesia dengan berbagai layanan, termasuk penyewaan kendaraan, logistik, lelang otomotif, dan jual beli kendaraan daring. Perusahaan ini memiliki 18 cabang di seluruh Indonesia dan didukung oleh Departemen Audit Internal yang bertugas mengevaluasi kepatuhan, efektivitas, serta pencegahan fraud. Salah satu fokus penelitian ini adalah prosedur opname audit aset non-unit, seperti meja dan kursi, yang masih memiliki beberapa kelemahan. Oleh karena itu, penelitian ini bertujuan merancang aplikasi berbasis kode QR untuk meningkatkan efisiensi audit aset non-unit. Pemilihan PT XYZ sebagai objek penelitian didasarkan pada pengalaman penulis sebagai auditor internal di perusahaan tersebut.

Penelitian ini mengikuti metode System Development Life Cycle (SDLC) dengan pendekatan Classical Waterfall, yang terdiri dari enam tahap utama: pengumpulan kebutuhan melalui wawancara dengan auditor internal, perancangan aplikasi, implementasi, pengujian, verifikasi, serta operasi dan pemeliharaan. Fokus utama penelitian adalah menganalisis dan merancang aplikasi audit aset non-unit hingga tahap desain dan pengujian, dengan harapan aplikasi ini dapat meningkatkan efektivitas dan akurasi dalam proses audit di PT XYZ.

3. HASIL DAN PEMBAHASAN

Tahap Analisis

Studi Kelayakan (Feasibility Study)

Feasibility study bertujuan untuk meninjau apakah terdapat solusi layak terhadap permasalahan yang ada, serta menentukan apakah layak secara finansial dan teknis mengenai pengembangan perangkat lunak yang akan dirancang (Prabowo, 2020). Melalui tahap wawancara dengan salah satu anggota tim audit internal, ditemukan beberapa poin masalah yang dihadapi pada prosedur opname audit aset non-unit, yaitu:

- A. Proses persiapan pre-opname audit yang memakan waktu auditor. Proses ini mengacu pada persiapan yang dilakukan auditor sebelum melaksanakan perjalanan menuju ke cabang penugasan, dimana auditor akan mempersiapkan

dan mencetak sebuah lembaran berisi daftar dari semua aset non-unit dari cabang penugasan, yang kemudian akan diceklis satu per satu ketika dilakukannya opname audit aset non-unit di cabang tersebut. Tahap ini dinilai kurang memberikan nilai tambah pada hasil akhir, dan waktu kerja dari auditor dapat dialokasikan pada aktivitas yang lebih bersifat analitis dan produktif.

- B. Proses pengisian ulang data rekapan hasil opname audit yang redundan. Ketika proses opname audit aset non-unit telah selesai dilakukan di cabang penugasan, maka langkah selanjutnya bagi auditor; jika mengacu pada alur opname audit tradisional; adalah melakukan perekapan ulang dari lembar berisi daftar aset non-unit yang telah diceklis menuju Excel, tentunya secara manual. Seperti poin sebelumnya, tahap ini tidak memberikan nilai tambah dan auditor harusnya dapat melakukan aktivitas yang lebih produktif.

Selain poin-poin diatas, sebagai sebuah prosedur yang cukup penting dalam menjaga kondisi dan keadaan dari aset perusahaan, ditambah juga dengan kemudahan dan tidak diperlukannya biaya signifikan dalam mengembangkan aplikasi ini, menjadikan pengembangan aplikasi pembantu sebagai solusi yang layak untuk meningkatkan efektivitas dan efisiensi prosedur opname aset non unit.

Analisis dan Spesifikasi Kebutuhan

Dalam sebuah prosedur opname aset non-unit, data yang dihasilkan dari kegiatan tersebut akan digunakan untuk menganalisa mengenai seberapa tingkat kepatuhan dari auditee atau dalam hal ini cabang operasional dalam menjaga kondisi aset tetap baik dan mempunyai label sesuai dengan ketentuan perusahaan. Sebagai langkah yang menjadi sumber data untuk analisa ini, tentunya dibutuhkan beberapa poin penting yang harus dipenuhi untuk menjaga kualitas dari audit. Pada penelitian ini, peneliti ingin berfokus pada peningkatan kecepatan dari prosedur opname aset dan pada akurasi informasi yang diambil pada pelaksanaan prosedur di lapangan.

Prosedur audit pada PT XYZ yang berkaitan secara khusus dengan opname audit aset non-unit berpaku kepada SOP yang ditetapkan oleh departemen General Affairs, sebuah departemen di Head Office yang memiliki tanggungjawab dalam pengelolaan aset non-unit di cabang maupun pada Head Office sendiri. Terdapat dua poin penting dari SOP tersebut yang menjadi acuan prosedur audit opname aset non-unit, yaitu:

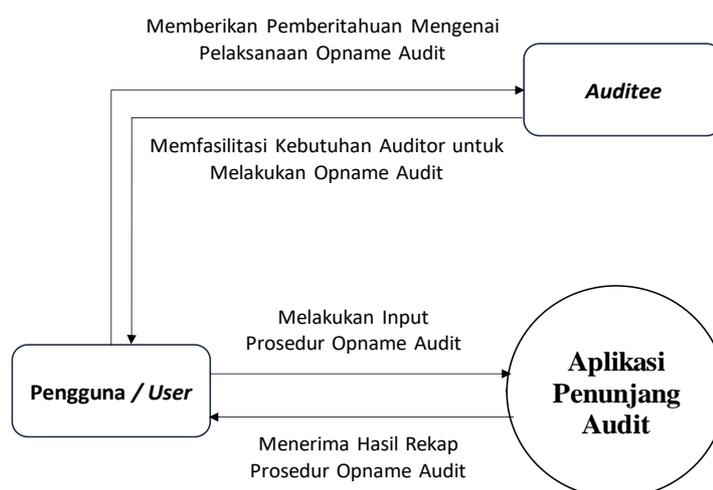
1. Kesesuaian aset non-unit berdasarkan informasi pada sistem dan pada lokasi.

2. Kondisi aset non-unit harus sesuai dengan SOP dari General Affairs, yaitu aset harus masih bisa digunakan secara wajar, dan masih tercantum secara jelas label aset non-unitnya.

Mengetahui kriteria yang ada pada SOP ini akan menjadi penting pada proses desain agar tercapai sebuah aplikasi penunjang opname aset non-unit yang layak guna. Melalui pemahaman kriteria, maka dalam setiap langkah desain aplikasi akan sesuai dengan tujuan awal prosedur opname aset non-unit dan setiap kriteria yang dibutuhkan akan dapat difasilitasi dengan baik melalui aplikasi

Perancangan Aplikasi

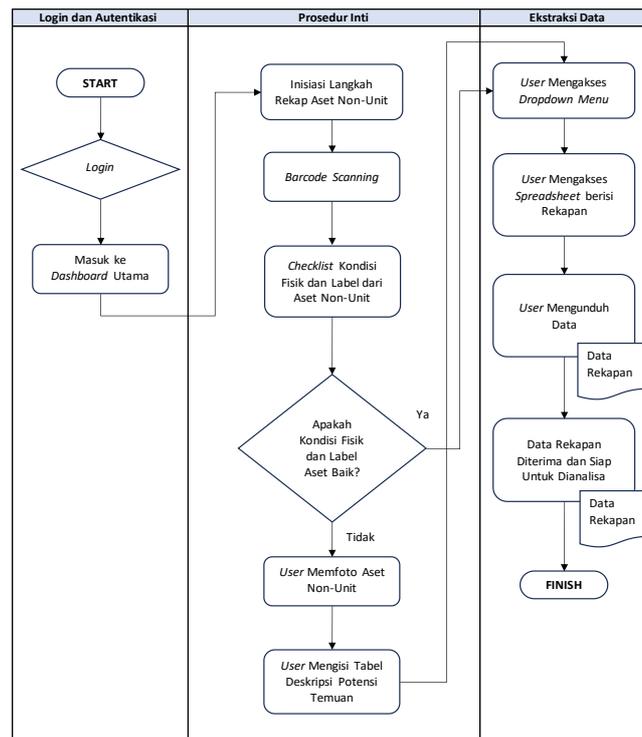
Data Flow Diagram merupakan sebuah grafik yang menggambarkan cara Bergeraknya alur informasi dan data dari *input* menuju *output* ([Hasanah et al, 2020](#)). Berikut adalah skema DFD dari aplikasi penunjang opname audit aset non-unit.



Gambar 1 Data Flow Diagram

Auditor internal sebagai pengguna, berinteraksi dengan *auditee* untuk memberikan pemberitahuan mengenai pelaksanaan opname audit aset non-unit. Pemberitahuan ini dibutuhkan agar *auditee* dapat memfasilitasi kebutuhan auditor internal dalam melaksanakan opname audit. Interaksi auditor internal dengan aplikasi penunjang audit adalah untuk melakukan input yang dibutuhkan untuk prosedur opname audit, dan aplikasi penunjang audit akan menyimpan data hasil rekap yang kemudian dapat diunduh oleh auditor internal.

Sequence Diagram diperlukan untuk menunjukkan komunikasi dinamis antar objek selama eksekusi dari sebuah prosedur (Prabowo, 2020). Berikut adalah *Sequence Diagram* dari aplikasi pembantu audit yang telah dirancang, dibagi mulai dari tahap awal yaitu proses login dan autentikasi, lalu prosedur inti, diakhiri dengan langkah ekstraksi data dengan tujuan untuk memudahkan visualisasi alur kerja dari aplikasi penunjang proses audit opname aset non-unit.



Gambar 2 Sequence Diagram

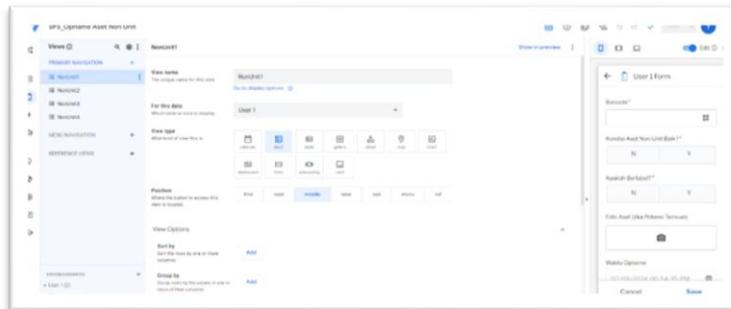
Berikut merupakan sebuah *flowchart* yang menggambarkan alur prosedur opname audit aset non-unit yang dilaksanakan melalui aplikasi penunjang audit. Jika dibandingkan dengan *flowchart* yang telah dipaparkan pada gambar 2, bisa dilihat bahwa melalui penggunaan aplikasi ini, tidak diperlukan proses persiapan pre-opname. Auditor dapat memulai proses opname dengan melakukan *login* pada *AppSheet*. Setelah proses *login* selesai, maka pengguna akan tiba pada tampilan utama dimana prosedur inti dapat dimulai.

Prosedur inti akan dimulai dengan pemindaian kode QR yang tertera pada label aset non-unit, dilanjutkan dengan pengisian beberapa menu yang akan dijelaskan pada subbab-subbab selanjutnya. Ketika proses opname audit rampung, auditor juga tidak perlu melakukan rekap ulang karena data telah tersimpan pada *Google Spreadsheets*,

yang kemudian dapat diunduh dalam bentuk .xlsx dan dianalisis lebih lanjut oleh auditor internal.

App Editor

Sebagai sebuah platform LCDP, desain *AppSheet* dilakukan melalui *app editor*, dimana pengguna, khususnya pemilik aplikasi dapat melakukan modifikasi dan perancangan aplikasi melalui pilihan-pilihan yang diberikan. Berikut adalah contoh tampak *app editor AppSheet*.

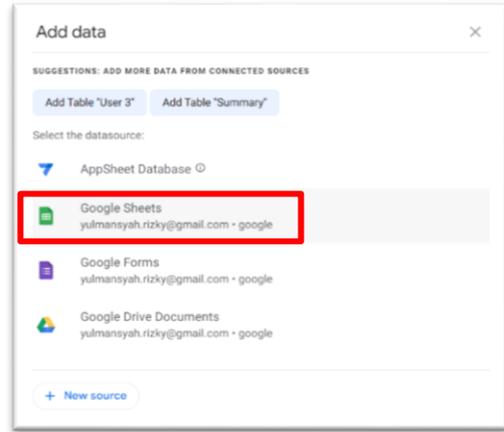
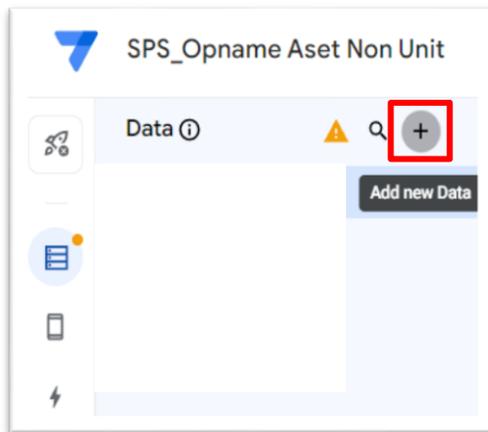


Gambar 3 Antarmuka App Editor

A. Impor Sumber Data

Langkah pertama yang dapat dilakukan adalah dengan impor sumber data yang akan menjadi basis data atau tempat disimpannya data dari hasil rekap yang dilakukan melalui aplikasi. Pada pengembangan aplikasi ini, penulis menggunakan *Google Spreadsheet* sebagai pilihan basis data karena familiaritas yang tinggi dengan *Microsoft Excel*, serta data rekap dapat diunduh dengan format “xlsx.” (*Excel*) yang akan memudahkan auditor dalam pengunduhan data serta analisis data.

Setelah basis data telah disiapkan dalam bentuk *Google Spreadsheet*, maka pengembang dapat menekan tombol “*Add New Data*” untuk menyambungkan basis data

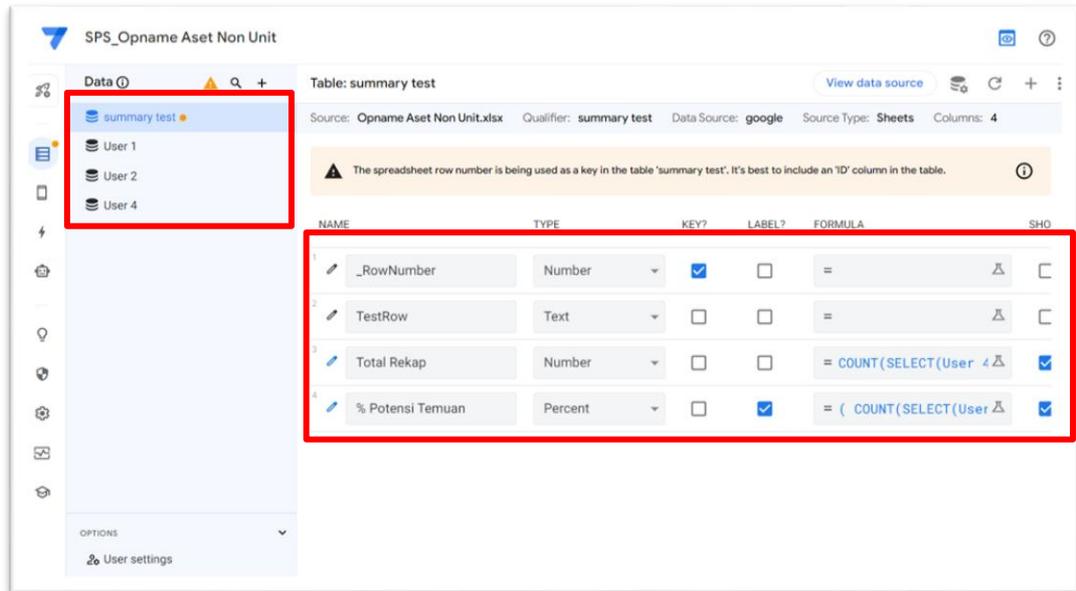


dengan *AppSheet* agar proses-proses selanjutnya dapat dilakukan.

Gambar 4 Tombol *Add New Data*

Gambar 5 Pilihan Basis Data

Ketika langkah sebelumnya selesai dilaksanakan dan basis data telah tersambung dengan *AppSheet*, maka akan muncul secara otomatis setiap *sheet* yang ada



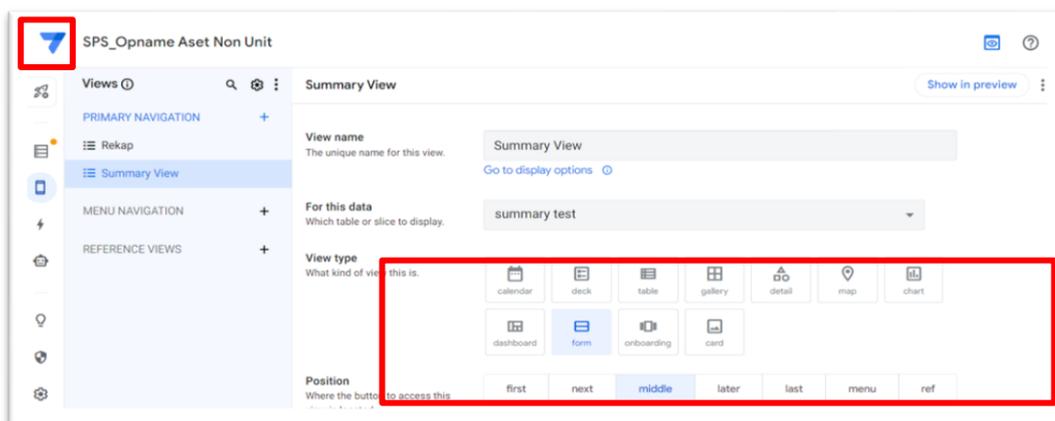
pada basis data sebagai tabel yang akan dapat dimodifikasi melalui *AppSheet*.

Gambar 4.3.2 Kolom Terhubung di *App Editor*

B. Manajemen View

View dalam *Google AppSheet* adalah cara penampilan data yang ada di aplikasi melalui antarmuka (*interface*) pengguna yang akan secara reguler diakses oleh pengguna aplikasi. Setiap *view* akan dapat dihubungkan dengan *sheet* yang dipilih pada basis data yang dipakai. Untuk manajemen *view* yang ada pada aplikasi, maka pengembang dapat mengakses bagian *view* pada sisi kiri *App Editor*.

Pengembang akan menghubungkan setiap *view* dengan tabel atau *sheet* yang diinginkan, lalu memberikan penamaan yang sesuai dengan prosedur yang dijalankan. Untuk kali ini, karena prosedur opname aset non-unit meliputi kegiatan perekapan, maka akan dinamakan “Rekap”, dan data akan terhubung ke basis data yang berhubungan dengan kegiatan perekapan data aset non-unit. Pada *view* kedua, penulis akan menggunakan *view* ini sebagai ringkasan dari data rekap yang akan dinamakan “*Summary View*”, dan akan dihubungkan dengan data atau *sheet* yang berkaitan dengan ringkasan data rekap.

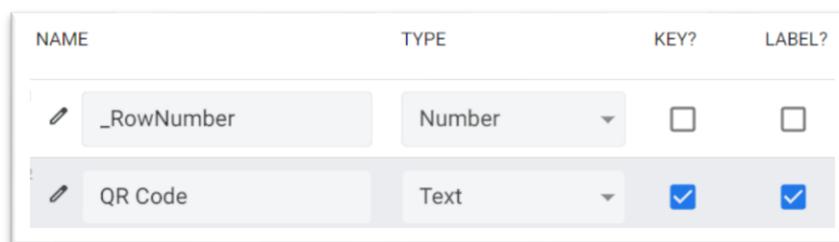


Gambar 5 *Summary View* Terhubung di *App Editor*

C. Modifikasi Menu Utama

Pemindaian QR Code

Sebagai langkah pertama pada prosedur inti sesuai dengan *flowchart* subbab 4.3, kolom *QR Code* menjadi kolom pertama yang akan digunakan pada prosedur inti. Tujuan utama dari kolom ini adalah untuk menarik data berbentuk teks yang tersimpan pada *QR Code*. Untuk menghindari pemindaian berulang, maka juga harus ada fitur pencegah duplikat pada langkah ini. Hal tersebut dapat dilakukan dengan mengubah tipe kolom menjadi “Text” agar bisa mengakomodasi teks yang akan ditarik dari *QR Code*, dan mencentang opsi “KEY?” dan “LABEL?” agar tidak terjadi duplikat data.



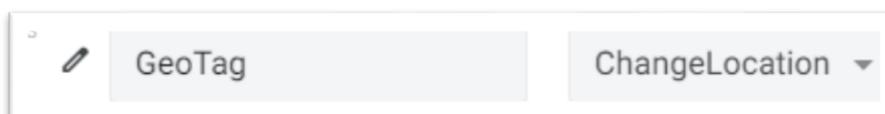
NAME	TYPE	KEY?	LABEL?
<input type="text" value="_RowNumber"/>	Number	<input type="checkbox"/>	<input type="checkbox"/>
<input type="text" value="QR Code"/>	Text	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Gambar 6 Pilihan Opsi Menu

Karena kolom ini termasuk dalam prosedur inti, maka kolom “SHOW?”, “EDITABLE?”, “REQUIRE?” akan dicentang untuk memunculkan kolom, memberikan kemampuan untuk mengisi kolom, dan memastikan bahwa pada setiap input, kolom ini harus selalu terisi.

GeoTag

Fitur *GeoTag* merupakan sebuah fitur yang akan mencantumkan lokasi gawai yang sedang dipakai dalam bentuk koordinat. Fungsi ini dapat dinyalakan dengan mengganti tipe kolom menjadi “*ChangeLocation*”.

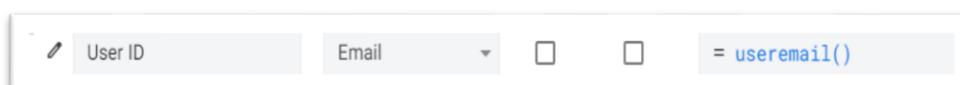


Gambar 7 Tipe Kolom *GeoTag*

Karena kolom ini tidak termasuk pada prosedur inti dan bersifat otomatis, maka kolom yang dicentang hanyalah kolom “EDITABLE?” karena auditor tidak perlu berinteraksi secara langsung dengan kolom *GeoTag*.

User ID

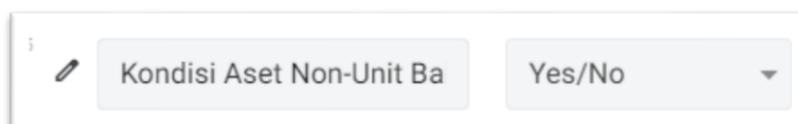
Kolom *User ID* memiliki fungsi yaitu untuk mengekstrak secara otomatis nama dari akun *email* yang sedang melakukan prosedur inti. Fungsi ini dapat digunakan dengan mengganti tipe kolom menjadi “*Email*”, yang akan mengekstrak *email* dengan rumus “=useremail()”. Kolom ini tidak perlu dicentang pada bagian “SHOW?” karena merupakan sebuah kolom yang berfungsi secara otomatis.



Gambar 8 Tipe Kolom *UserID*

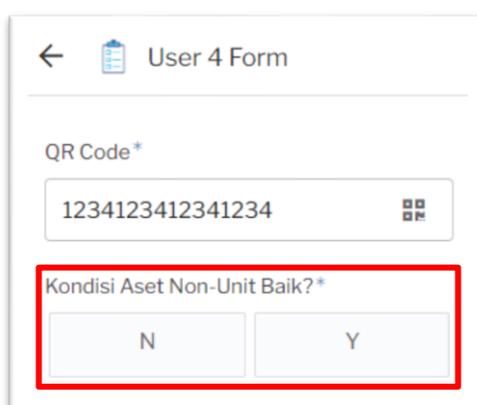
Checklist Kondisi Aset

Kolom ini berfungsi sebagai *checklist* yang dapat dipilih oleh auditor mengenai kondisi aktual dari aset non-unit. Dengan mengubah tipe kolom menjadi “Yes/No”, maka auditor akan dapat mencentang kondisi aset sesuai dengan kondisi aset tersebut di lapangan. Sebagai salah satu bagian dari prosedur inti, maka kolom ini harus diisi setiap kali rekap dilaksanakan. Berikut adalah tampak kolom pada *App Editor*.



Gambar 8 Tampak Kolom Kondisi Aset pada *App Editor*

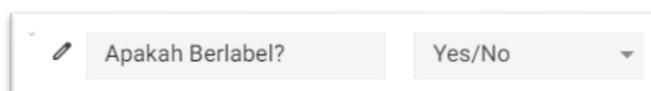
Berikut adalah tampak kolom tersebut dalam aplikasi ketika prosedur inti dilaksanakan.



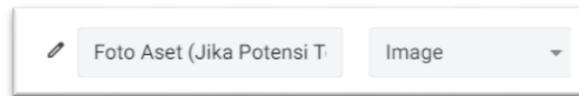
Gambar 9 Tampak Kolom Kondisi Aset pada Aplikasi

Checklist Label Aset

Kolom ini berfungsi sebagai *checklist* yang dapat dipilih oleh auditor mengenai adanya label aset non-unit pada aset yang sedang diperiksa. Dengan mengubah tipe kolom menjadi “Yes/No”, maka auditor akan dapat mencentang kondisi aset sesuai dengan kondisi label aset di lapangan. Sebagai salah satu bagian dari prosedur inti, maka kolom ini harus diisi setiap kali rekap dilaksanakan. Berikut adalah tampak kolom pada *App Editor*.



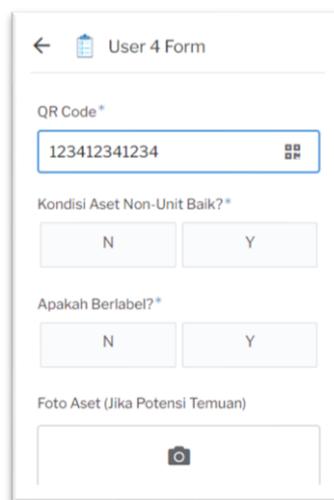
Gambar 10 Tampak Kolom Label pada *App Editor*



Gambar 11 Tampak Kolom Label pada Aplikasi

Foto Aset

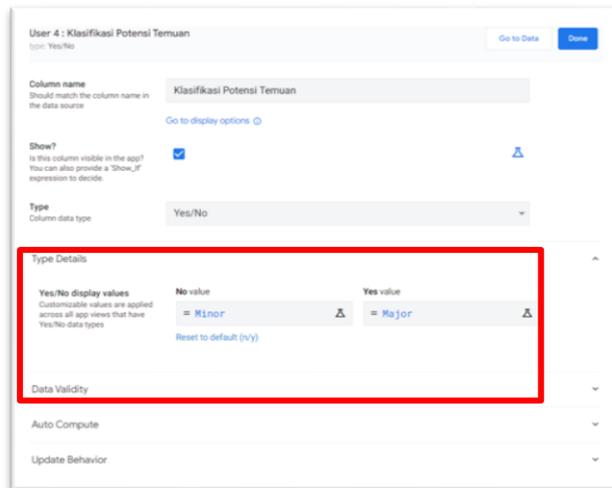
Kolom ini akan berfungsi sebagai input dalam bentuk foto yang akan tersimpan secara lokal dalam gawai auditor ketika ditemukannya aset non-unit yang berpotensi sebagai temuan. Untuk mengaplikasikan fitur ini, pengembang dapat mengubah tipe kolom menjadi “Image”. Berikut adalah tampak kolom pada *App Editor*.



Gambar 12 Tampak Kolom Foto Aset pada App Editor

Klasifikasi Potensi Temuan

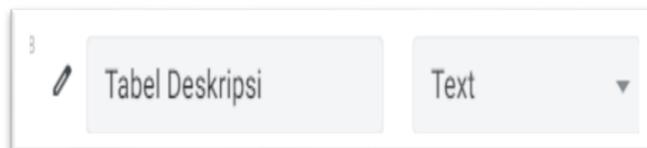
Kolom ini berfungsi sebagai menu yang akan digunakan oleh auditor untuk menentukan tingkat temuan dalam dua klasifikasi yaitu “Minor” untuk potensi temuan yang tidak memerlukan tindak lanjut secara langsung dan “Major” untuk potensi temuan yang harus segera ditindaklanjuti. Menu ini dapat dibuat dengan cara memberikan tipe menu dalam bentuk “YES/NO”. Tahap selanjutnya adalah mengganti *display value* dari kolom menjadi “Minor” dan “Major”. Berikut adalah tampilannya.



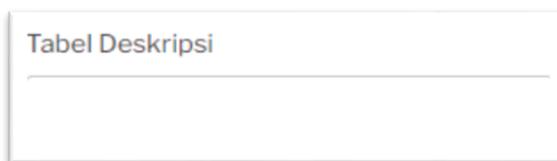
Gambar 13 Tampak Kolom Klasifikasi pada App Editor

Tabel Deskripsi

Kolom ini berfungsi sebagai tabel deskripsi berbentuk teks yang akan diisi oleh auditor ketika terdapat potensi temuan. Dengan mengubah tipe kolom menjadi "Text", maka kolom dapat diisi dengan teks deskriptif. Berikut adalah tampilannya.



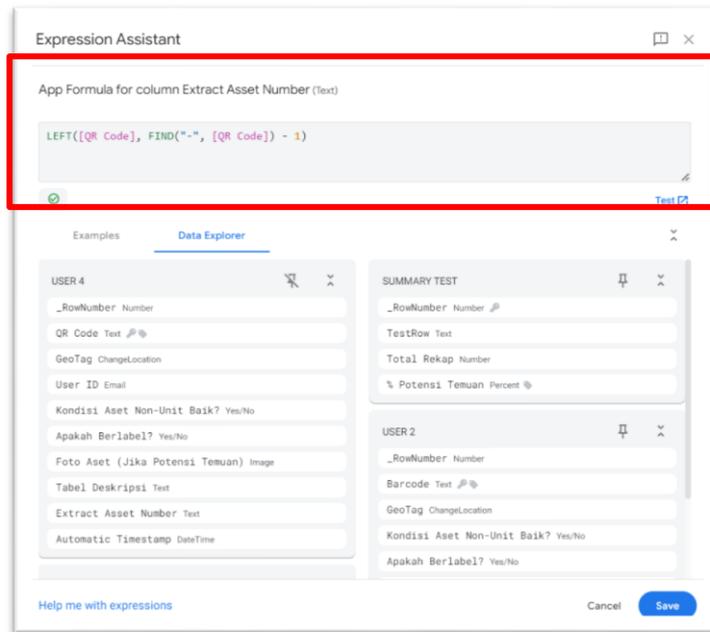
Gambar 14 Tampak Kolom Tabel Deskripsi pada App Editor



Gambar 15 Tampak Kolom Tabel Deskripsi pada Aplikasi

Extract Asset Number

Kolom ini memiliki fungsi untuk mengekstrak dua belas angka pertama dari teks yang telah ditarik dari QR Code pada langkah pertama yaitu pemindaian QR Code. Tahap pertama yang dapat dilakukan adalah mengubah tipe kolom menjadi "Text" yang akan mengakomodasi hasil teks yang diekstrak dari QR Code. Langkah kedua adalah dengan mengisi tabel formula dengan formula yang sesuai.



Gambar 16 Pengisian App Formula

Berikut ini adalah contoh *string* yang diekstrak dari *QR Code* yang ada pada label aset non-unit:

60000003531-LAPTOP-LENOVOIDEAPAD-HEADOFFICE-BLOSSOM MULTIMEDIA-1234-A.n Fulan, Analyst Internal Audit HO-

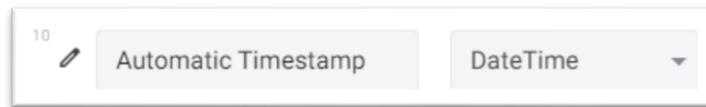
Pada contoh ini, dua belas angka pertama merupakan nomor aset yang ingin diekstrak oleh kolom *extracted asset number*. Berikut adalah rumus yang diaplikasikan pada kolom *extracted asset number* untuk mengambil dua belas angka pertama tersebut.

“LEFT([QR Code], FIND("-", [QR Code]) - 1)”

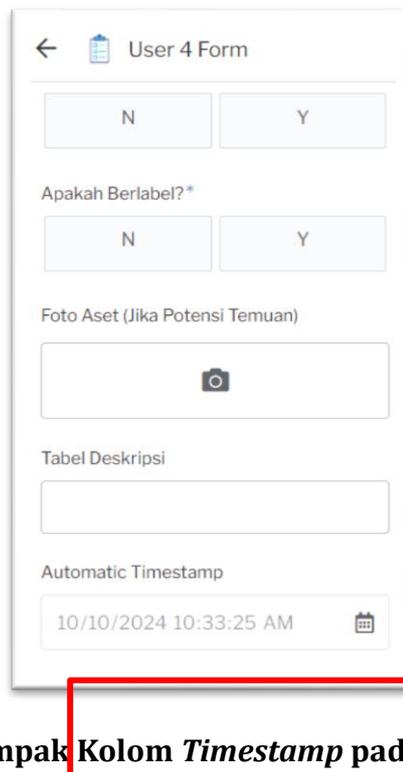
Bagian “FIND("-", [QR Code])” bekerja untuk mencari “-” pertama pada *cell* kolom *QR Code*. Pada *string* contoh tersebut, posisi “-” ada pada urutan ke-13. Ditambahkannya “-1” pada “FIND("-", [QR Code])” dilakukan untuk mengurangi awal dari urutan pencarian yang awalnya urutan ke-13 menjadi ke-12. Rumus “LEFT” lalu akan mengekstrak dua belas karakter pertama dalam *string* tersebut, yang merupakan nomor aset non-unit yang dicari.

Automatic Timestamp

Kolom ini berfungsi sebagai *timestamp* yang akan bekerja secara otomatis setiap kali inisiasi prosedur inti dilakukan. Langkah yang dapat dilakukan adalah dengan mengganti tipe kolom menjadi “*DateTime*”. Berikut adalah tampilannya pada *App Editor* dan juga pada aplikasi.



Gambar 17 Tampak Kolom *Timestamp* pada *App Editor*



Gambar 18 Tampak Kolom *Timestamp* pada Aplikasi

Modifikasi Summary View

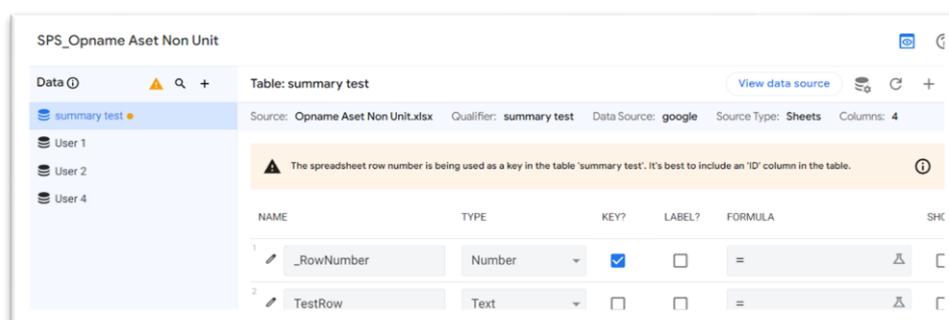
Komposisi dari *Summary View* akan diisi dengan dua menu yang akan dibuat melalui *virtual column*. Seperti yang sudah dibahas sebelumnya, *virtual column* digunakan untuk data yang bersifat temporer seperti kalkulasi, dan pada *view* ini, akan digunakan untuk kalkulasi ringkasan dari kegiatan opname audit.

Dummy Column

Dalam *AppSheet*, sebuah *view* tidak bisa hanya terdiri atas *virtual column* saja. Karenanya, harus dibuat sebuah kolom kosong yang tidak digunakan pada basis data, dan untuk pilihan “SHOW?” tidak dicentang agar menu tersebut tidak dapat dilihat serta diakses dalam aplikasi.

Total Rekap

Tujuan dari kolom total rekap adalah memberikan total dari seluruh rekap yang sudah masuk pada basis data, atau *Google Spreadsheet*. Data ini bersifat temporer dan akan secara konstan terbaru setiap kali prosedur opname audit diselesaikan, dan karenanya akan menggunakan *virtual column*. Berikut adalah cara membuat sebuah *virtual column* baru.



Gambar 19 Tampak Kolom Total Rekap pada App Editor

Rumus yang dibutuhkan untuk menghitung total rekap pada basis data disimpannya hasil rekap adalah rumus yang dapat menghitung jumlah kolom terisi pada kolom spesifik, dan pada kasus ini, kolom tersebut merujuk pada kolom “QR Code” pada *sheet* disimpannya data rekap. Kolom ini dipilih karena sebagai langkah pertama dalam prosedur inti yang tidak bisa dilewatkan oleh auditor, kolom ini akan selalu terisi sehingga konsistensi rumus akan terjaga. Berikut adalah rumus yang diberikan pada kolom ini.

“COUNT(SELECT(User 4[QR Code], ISNOTBLANK([QR Code])))”

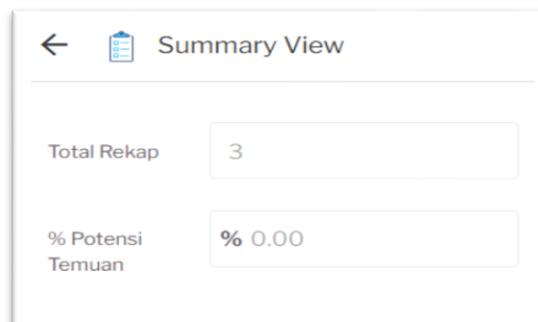
Seperti yang telah dijelaskan, rumus ini menghitung jumlah entri pada *sheet* “QR Code” dengan menghitung jumlah kolom yang tidak kosong.

Persentase Potensi Temuan

Pada kolom ini, akan diberikan sebuah tampilan yang menunjukkan persentase potensi temuan dari seluruh entri rekap opname audit. Tampilan ini akan secara berkala terbaru seiring dengan masuknya data rekap baru pada basis data. Langkah pertama yang dapat dilakukan adalah dengan membuat *virtual column* baru untuk mengakomodasi rumus yang dapat menghitung persentase tersebut. Berikut adalah rumus yang digunakan dalam kolom ini.

“(COUNT(SELECT(User 4[Foto Aset (Jika Potensi Temuan)], ISNOTBLANK([Foto Aset (Jika Potensi Temuan)]))) / COUNT(SELECT(User 4[QR Code], ISNOTBLANK([QR Code]))))”

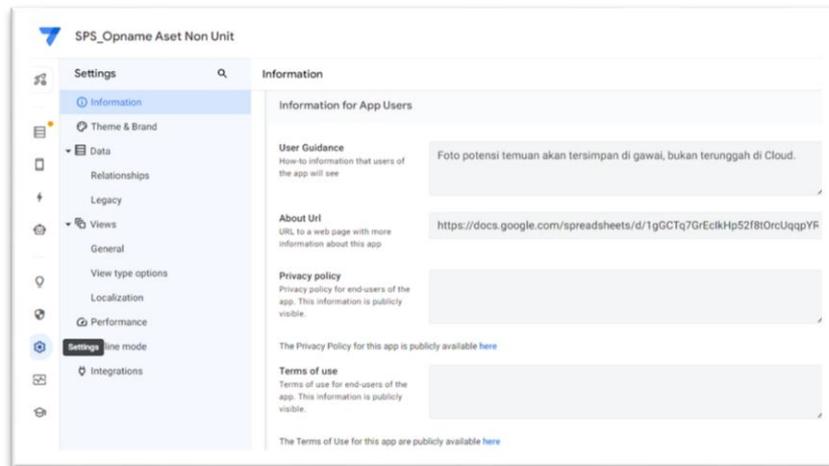
Melalui rumus ini, dapat ditemukan perbandingan antara baris yang terisi pada kolom “Foto Aset (Jika Potensi Temuan)” dengan baris yang terisi pada kolom “QR Code”. Dengan mengetahui jumlah berapa foto aset yang sudah masuk pada basis data, dan membandingkannya dengan jumlah baris terisi pada kolom “QR Code” maka didapatkan persentase potensi temuan. Dua kolom ini digunakan karena pada kolom “QR Code”, seperti yang telah dijelaskan sebelumnya, sudah pasti terisi dan dapat diandalkan sebagai salah satu variabel rumus. Kolom foto aset digunakan karena setiap kali ada temuan, foto yang diambil oleh auditor akan tetap tercatat jejaknya walaupun foto tersebut tersimpan secara lokal pada gawai auditor. Berikut adalah tampilan *summary view* yang sudah diselesaikan.



Gambar 20 Tampak Kolom Persentase Potensi Temuan pada Aplikasi

Modifikasi Dropdown Menu

Dropdown menu merupakan sebuah menu yang dapat diakses dari layar utama aplikasi. Pada *menu* ini, akan disediakan sebuah langkah untuk mengakses data hasil rekapian auditor. Untuk langkah tersebut, penulis menggunakan opsi *About* sebagai tempat dimana auditor dapat mengakses *Google Spreadsheet* atau basis data rekapian melalui sebuah *hyperlink*. Informasi yang muncul pada opsi *About* dapat dimodifikasi melalui bagian *Settings*, dan pada topik *Information*. Berikut adalah tampilannya pada *App Editor*.



Gambar 21 Modifikasi Opsi *About* pada *Dropdown Menu*

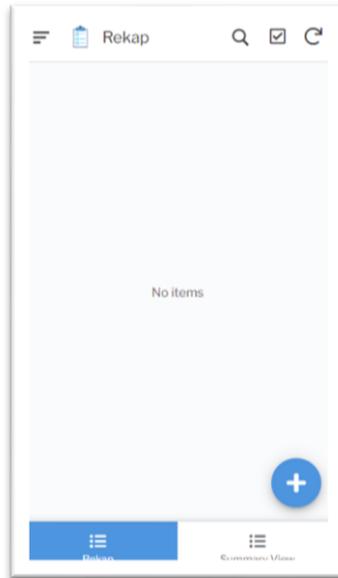
Setelah langkah ini dilaksanakan, maka tautan tersebut ketika diakses melalui opsi “*About*” pada *dropdown menu* akan langsung membawa pengguna ke lokasi basis data disimpan yang dapat secara langsung diunduh dan dianalisa lebih lanjut.

Proses Penggunaan Aplikasi

Proses Login dan Autentikasi

Proses opname audit aset non-unit akan dimulai dengan proses dimana pengguna akan melakukan *login* atau bergabung di aplikasi. Langkah autentikasi ini diperlukan untuk menghindari adanya akses masuk dari pengguna selain dari tim auditor internal. Karena *AppSheet* merupakan salah satu produk *Google*, proses login ini secara standar dilakukan melalui *Google* sendiri sebagai *Authentication Provider* walaupun terdapat juga banyak opsi lain yang bisa dipilih melalui *App Editor*. Fitur *login* dapat difungsikan dengan menggunakan fitur “*Require User Sign-in*” pada bagian *Security* di *App Editor*.

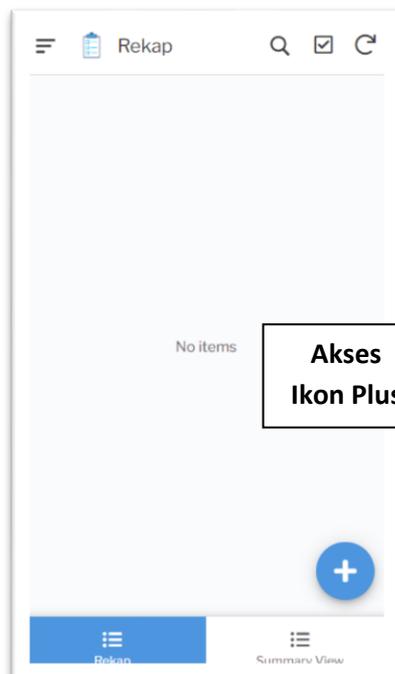
Setelah proses login berhasil dilaksanakan, maka auditor akan berada pada dasbor utama aplikasi. Berikut adalah tampilan dasbor tersebut ketika diakses melalui *smartphone*.



Gambar 21 Tampak Dasbor Utama

Prosedur Inti

Prosedur inti merupakan sebuah bagian dimana auditor dapat mulai melakukan *input* data opname aset non-unit. Sebagai bentuk Proses ini dapat dimulai dengan



menekan ikon plus (+) pada bagian kanan bawah.

Gambar 22 Letak Akses

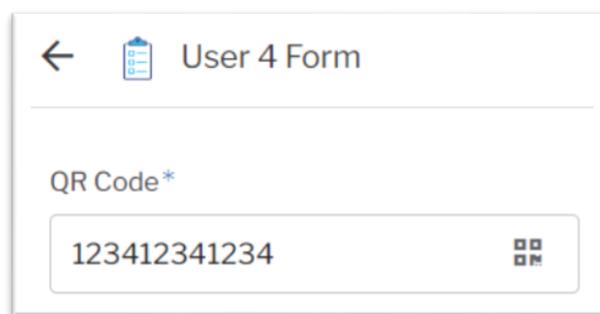
Ikon Plus

Ketika ikon plus ditekan, maka kamera akan teraktifkan secara otomatis dan langkah selanjutnya yaitu proses pemindaian *QR Code* dapat dilakukan. Berikut adalah

tampak aplikasi setelah ikon plus ditekan dan proses pemindaian *QR Code* dilaksanakan, diikuti dengan penjelasan dari setiap interaksi auditor pada aplikasi penunjang audit opname aset non-unit.

Pemindaian QR Code

Langkah pemindaian *QR Code* menjadi tahap pertama dalam prosedur inti aplikasi penunjang audit. *QR Code* yang dimaksud dalam tahap ini adalah *QR Code* yang terlampir pada setiap label aset non-unit. Pemindaian *QR Code* dilakukan dengan tujuan untuk menarik informasi yang tersimpan dalam *QR Code* tersebut sehingga dapat diproses pada tahap-tahap selanjutnya.

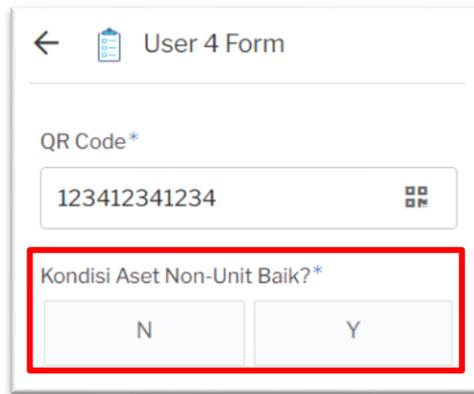


Gambar 23 Tampak Kolom *QR Code* Pada Aplikasi

Penggunaan *QR Code* ini juga merupakan bentuk dari *data capture control* dan *error control*. Dengan mengimplementasikan langkah pertama dengan pemindaian *QR Code*, dipadukan dengan pencegahan duplikat, informasi dari *QR Code* akan diekstrak secara otomatis, dan memperkecil adanya potensi kesalahan jika dibandingkan dengan proses *input* secara manual.

Checklist Kondisi Aset Non-Unit

Tahap yang dilakukan setelah auditor melakukan pemindaian *QR Code* adalah pengisian *checklist* kondisi aset non-unit. Sesuai dengan SOP prosedur opname aset non-unit, diperlukan pengecekan fisik oleh auditor terhadap kondisi tiap aset non-unit.

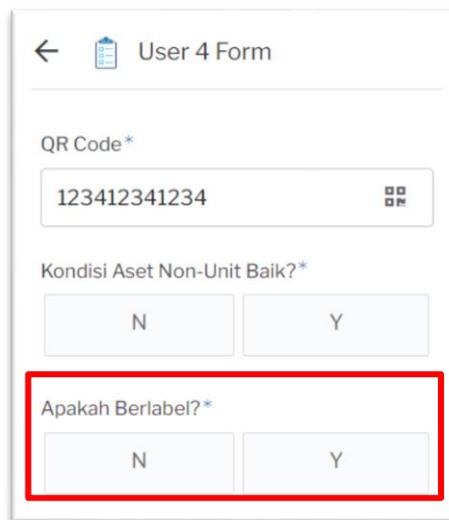


Gambar 24 Tampak Kolom Kondisi Aset Pada Aplikasi

Jika kondisi dari aset sudah sesuai kriteria yang ditetapkan, maka auditor dapat memilih Y, dan jika kondisi aset tidak sesuai kriteria, maka sebaliknya auditor dapat memilih N.

Checklist Ketersediaan Label

Mengikuti tahap sebelumnya yang mengacu pada SOP prosedur opname aset non-unit, pada tahap ini auditor dapat melakukan pengecekan secara fisik terhadap keadaan label.

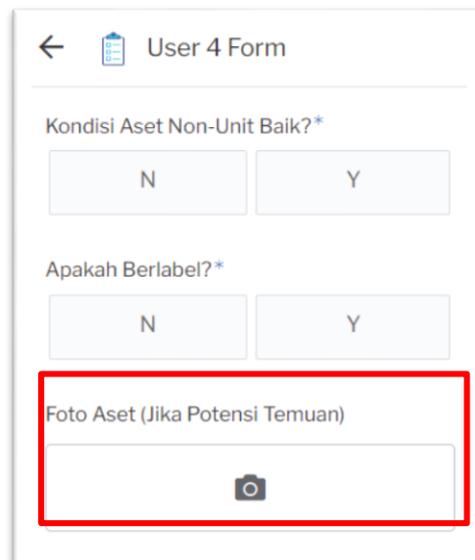


Gambar 25 Tampak Kolom Label Pada Aplikasi

Jika keadaan label aset non-unit masih baik, dengan artian masih bisa dibaca dengan jelas dan tidak terdapat goresan pada informasi penting label, maka auditor dapat memilih Y, dan sebaliknya jika kondisi label buruk, maka auditor dapat memilih N.

Foto Aset Non-Unit Potensi Temuan

Ketika auditor menemukan ketidaksesuaian dalam dua tahap sebelumnya, maka dapat disimpulkan bahwa terdapat potensi temuan atas aset non-unit spesifik tersebut. Sesuai dengan prosedur opname aset non-unit, setiap potensi temuan akan dikomunikasikan terlebih dahulu dengan manajemen dengan tujuan untuk mengetahui alasan mengapa sebuah aset non-unit tertentu dapat digolongkan sebagai potensi temuan.

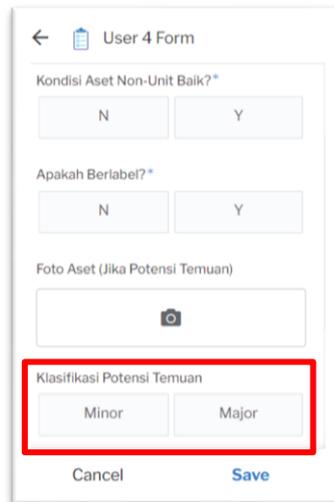


Gambar 25 Tampak Kolom Foto Aset Pada Aplikasi

Dengan adanya foto dari aset yang berkategori potensi temuan, maka akan lebih mudah bagi auditor untuk memasukkan bukti ke laporan audit dan juga sebagai informasi tambahan yang dapat digunakan selanjutnya.

Klasifikasi Potensi Temuan

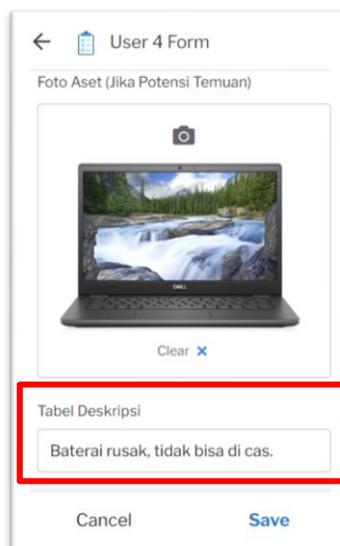
Pada tahap ini, auditor dapat memilih jika potensi temuan yang diperiksa perlu penindakan lanjut segera atau memiliki prioritas lebih rendah untuk ditindaklanjuti dengan memilih “*Minor*” dan “*Major*”. Berikut adalah tampilannya.



Gambar 26 Tampak Klasifikasi Potensi Temuan Pada Aplikasi

Tabel Deskripsi Aset Potensi Temuan

Tabel deskripsi ini berfungsi sebagai informasi tambahan yang akan dibutuhkan oleh auditor untuk mendeskripsikan apa alasan yang menjadikan sebuah aset non-unit berpotensi temuan.



Gambar 27 Tampak Kolom Tabel Deskripsi Pada Aplikasi

4. KESIMPULAN

Dengan memulai proses pengembangan aplikasi sesuai dengan metode *classical waterfall* yaitu melalui *requirement gathering & analysis* yang dilakukan dengan metode wawancara, dan dilanjutkan dengan proses perancangan alur perangkat lunak secara keseluruhan dengan memastikan bahwa poin-poin penting yang harus dapat dipenuhi

oleh aplikasi penunjang audit dapat terpenuhi, sebuah rancangan aplikasi penunjang prosedur opname audit non-unit dapat terancang dengan baik. Berdasarkan perancangan aplikasi yang telah dilakukan, serta melalui komunikasi dengan *user* berkaitan dengan projek aplikasi yang telah rampung, dapat disimpulkan dari kedua hal tersebut bahwa aplikasi penunjang audit sudah memenuhi kebutuhan yang diharapkan dan mampu mengakomodasi kebutuhan auditor internal berkaitan dengan prosedur opname audit aset non-unit secara efektif. Melalui penggunaan aplikasi ini, masalah redundansi yang ditemui pada alur audit opname aset non-unit tradisional dan juga perekapan ulang data hasil dapat dieliminasi, dan tujuan penelitian dapat tercapai.

5. DAFTAR PUSTAKA

- Prabowo, M., 2020. Metodologi pengembangan sistem informasi. LP2M Press IAIN Salatiga.
- Hastings, N.A., 2010. Physical asset management (Vol. 2, pp. 209-221). London: Springer.
- Zamzami, F. and Faiz, I.A., 2018. Audit Internal: Konsep Dan Praktik. Ugm Press.
- Lois, P., Drogalas, G., Karagiorgos, A. and Tsikalakis, K., 2020. Internal audits in the digital era: opportunities risks and challenges. *EuroMed Journal of Business*, 15(2), pp.205-217.
- Messier Jr, W.F., Glover, S.M. and Prawitt, D.F., 2017. Auditing & assurance services: A systematic approach. McGraw-Hill.
- Rao, S., Ahmad, A., Horsman, W. and Kaptein-Russell, P., 2001. The importance of innovation for productivity. CSLS.
- Hoos, E., Gröger, C., Kramer, S. and Mitschang, B., 2014. Improving Business Processes through Mobile Apps. proceedings of the 16th International Conference on Enterprise Information Systems (ICEIS), pp.71-82.
- Albrecht, W.S., Albrecht, C.O., Albrecht, C.C. and Zimbelman, M.F., 2012. Fraud examination (p. 696). New York, NY: Thomson South-Western.
- Davidson, B.I., Desai, N.K. and Gerard, G.J., 2013. The effect of continuous auditing on the relationship between internal audit sourcing and the external auditor's reliance on the internal audit function. *Journal of Information Systems*, 27(1), pp.41-59.
- Abdollahzadegan, A., Che Hussin, A.R., Moshfegh Gohary, M. and Amini, M., 2013. The organizational critical success factors for adopting cloud computing in SMEs. *Journal of Information Systems Research and Innovation (JISRI)*, 4(1), pp.67-74.

- Murti, N.L.B.M., Rahardjo, K. and Riza, M.F., 2013. Pengaruh Motivasi Dan Disiplin Terhadap Efektivitas Kerja Karyawan. *Jurnal Administrasi Bisnis (JAB)*, 6(2).
- Indrayani, H., 2012. Penerapan teknologi informasi dalam peningkatan efektivitas, efisiensi dan produktivitas perusahaan. *Jurnal El-Riyasah*, 3(1), pp.48-56.
- Syam, S., 2020. Pengaruh Efektifitas Dan Efisiensi Kerja Terhadap Kinerja Pegawai Pada Kantor Kecamatan Banggae Timur. *Jurnal Ilmu Manajemen Profitability*, 4(2), pp.128-152.
- Hartadi, M.G., Swandi, I.W. and Mudra, I.W., 2020. Warna dan prinsip desain user interface (ui) dalam aplikasi seluler “bukaloka”. *Jurnal Dimensi DKV: Seni Rupa dan Desain*, 5(1), pp.105-119.
- Rakestraw, T.L., Eunni, R.V. and Kasuganti, R.R., 2013. The mobile apps industry: A case study. *Journal of Business Cases and Applications*, 9, p.1.
- Sahay, A., Indamutsa, A., Di Ruscio, D. and Pierantonio, A., 2020, August. Supporting the understanding and comparison of low-code development platforms. In *2020 46th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)* (pp. 171-178). IEEE.
- Petrović, N., Radenković, M. and Nejković, V., 2020. Data-driven mobile applications based on AppSheet as support in COVID-19 crisis. *IcETran 2020*, pp.1-6.
- Bassil, Y., 2012. A simulation model for the waterfall software development life cycle. arXiv preprint arXiv:1205.6904.
- Pargaonkar, S., 2023. A Comprehensive Research Analysis of Software Development Life Cycle (SDLC) Agile & Waterfall Model Advantages, Disadvantages, and Application Suitability in Software Quality Engineering. *International Journal of Scientific and Research Publications (IJSRP)*, 13(08).
- Germain, É. and Robillard, P.N., 2005. Engineering-based processes and agile methodologies for software development: a comparative case study. *Journal of Systems and Software*, 75(1-2), pp.17-27.
- Fowler, M. and Highsmith, J., 2001. The agile manifesto. *Software development*, 9(8), pp.28-35.
- Coleman, G. and Verbruggen, R., 1998. A quality software process for rapid application development. *Software Quality Journal*, 7, pp.107-122.

- Mackay, H., Carne, C., Beynon-Davies, P. and Tudhope, D., 2000. Reconfiguring the user: Using rapid application development. *Social studies of science*, 30(5), pp.737-757.
- Daud, N.M.N., Bakar, N.A.A.A. and Rusli, H.M., 2010, June. Implementing rapid application development (RAD) methodology in developing practical training application system. In *2010 International Symposium on Information Technology* (Vol. 3, pp. 1664-1667). IEEE.
- Sadabadi, A.T. and Tabatabaei, N.M., 2009. Rapid prototyping for software projects with user interfaces. *Department of Computer Systems and Informatics Seraj Higher Education Institute, Iran*, št, 9, pp.85-90.
- Arens, A.A., Elder, R.J. and Beasley, M.S., 2012. *Auditing and assurance services: an integrated approach*. Prentice Hall.
- Louwers, T.J., Blay, A.D. and Sinason, D.H., 2018. *Auditing & assurance services*. McGraw-Hill.
- Feizizadeh, A., 2012. Strengthening internal audit effectiveness. *Indian Journal of Science and Technology*, pp.2777-2778.
- Coderre, D., 2009. *Internal audit efficiency through automation*. John Wiley & Sons, Inc.
- Lee, L., Whitworth, J. and Hermanson, S., 2015. The effects of information technology innovativeness on audit efficiencies. *Review of Business Information Systems*, 19(1).
- Hasanah, F.N. and Untari, R.S., 2020. *Buku Ajar Rekayasa Perangkat Lunak*. Umsida Press, pp.1-119.
- Mayselah, N., 2023. Optimalisasi Produktivitas dengan Metode Time and Motion Study di PT. XYZ. *Journal of Research on Industrial and System Engineering*, 1(1), pp.1-12.
- Sri Wahyuni, S.E., Dev, M.E., Rifki Khoirudin, S.E. and Dev, M.E., 2020. *Pengantar Manajemen Aset*. Nas Media Pustaka.
- Nuravita, A., Purnomo, W. and Pramono, D., 2020. Analisis dan Perancangan Sistem Informasi Audit Internal Berbasis Web (Studi pada PT Wijaya Karya Beton Tbk). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 4(10), pp.3291-3298.

- Wijoyo, A., Silalahi, A.R., Raihan, A., Arrasyid, P. and Diana, R., 2023. Sistem Informasi Manajemen Berbasis Cloud. *TEKNOBIS: Jurnal Teknologi, Bisnis dan Pendidikan*, 1(2).
- Chang, Y.H. and Ko, C.B., 2017. A study on the design of low-code and no code platform for mobile application development. *International journal of advanced smart convergence*, 6(4), pp.50-55.
- Oktavia, I.R., 2015. Peranan teknologi informasi dalam audit. *Bhirawa*, 2(2), pp.78-84.
- Amalia, D.R. and Laksmi, A.C., 2024, January. Implementation of fixed assets audit at PT U by KAP Y. In *Proceeding International Conference on Accounting and Finance* (pp. 335-342).
- Handayani, T., Furqon, A.H. and Supriyono, S., 2020. Rancang Bangun Sistem Inventori Pengendalian Stok Barang Berbasis Java Pada PT Kalibesar Artah Perkasa. *Jurnal SITECH: Sistem Informasi Dan Teknologi*, 3(1), pp.35-40.