



DELEGATE IT

# 2025 PORTFOLIO - EVIDENCE

EXPERIENCE & PROJECT  
EVIDENCE

**DATE :**

3 March 2025- 31 August 2025

**BY:**

Ruan Leenders

## Contents

|                                                                               |    |
|-------------------------------------------------------------------------------|----|
| <b>Section 1: Company Profile</b> .....                                       | 4  |
| <b>1.1 Overview:</b> .....                                                    | 4  |
| <b>1.1.1 Core Services:</b> .....                                             | 4  |
| <b>1.1.2 Value Proposition:</b> .....                                         | 4  |
| <b>1.2 Location:</b> .....                                                    | 4  |
| .....                                                                         | 6  |
| <b>Section 2: Projects</b> .....                                              | 7  |
| <b>2.1 SalesFront</b> .....                                                   | 7  |
| 2.1.1 About .....                                                             | 7  |
| 2.1.2 Market Opportunity .....                                                | 7  |
| 2.1.3 Technical Architecture .....                                            | 7  |
| 2.1.4 Software Features .....                                                 | 7  |
| 2.1.5 Target Market .....                                                     | 8  |
| 2.1.6 Insider Look .....                                                      | 9  |
| 2.1.7 My Contribution .....                                                   | 9  |
| 2.1.8 Multi-Tenant SaaS Architecture Implementation .....                     | 9  |
| 2.1.9 Advanced Sales Opportunity Management System .....                      | 10 |
| 2.1.10 Manufacturing Resource Planning (MRP) System .....                     | 11 |
| 2.1.11 Evidence Of Contribution .....                                         | 12 |
| 2.1.12 Development & Feature Implementation .....                             | 12 |
| 2.1.13 UI/UX & Mobile Enhancements .....                                      | 13 |
| 2.1.14 System Improvements & Optimizations .....                              | 13 |
| 2.1.15 Project Support, Testing & Research .....                              | 13 |
| 2.1.16 Bug Fixes & Status Tracking .....                                      | 13 |
| 2.1.17 Software Display .....                                                 | 14 |
| <b>2.2 PSA</b> .....                                                          | 17 |
| 2.2.1 About .....                                                             | 17 |
| 2.2.2 Market Opportunity .....                                                | 17 |
| 2.2.3 Technical Architecture .....                                            | 17 |
| 2.2.4 Software Features .....                                                 | 18 |
| 2.2.5 Target Market .....                                                     | 19 |
| 2.2.6 Insider Look .....                                                      | 19 |
| 2.2.7 My Contribution .....                                                   | 20 |
| 2.2.8 Mobile App Bootstrap, Firebase, and Service Composition .....           | 20 |
| 2.2.9 Secure API Client and AuthN/AuthZ Integration .....                     | 22 |
| 2.2.10 Nominations Workflow: Validation, Document Packaging, Submission ..... | 23 |
| 2.2.11 Voting Flow and Eligibility Rules .....                                | 24 |
| 2.2.12 Backend API Endpoints and Caching .....                                | 25 |
| 2.2.13 Evidence Of My Contribution .....                                      | 25 |
| 2.2.14 Development & Feature Implementation .....                             | 26 |
| 2.2.15 UI/UX & Mobile Enhancements .....                                      | 26 |

|                                                             |           |
|-------------------------------------------------------------|-----------|
| 2.2.16 System Improvements & Optimizations .....            | 26        |
| 2.2.17 Project Support, Testing & Research .....            | 27        |
| 2.2.18 Bug Fixes & Status Tracking .....                    | 27        |
| 2.2.19 Software Display .....                               | 27        |
| <b>2.3 MAN TCO .....</b>                                    | <b>31</b> |
| 2.3.1 About .....                                           | 31        |
| 2.3.2 Market Opportunity .....                              | 31        |
| 2.3.3 Technical Architecture .....                          | 32        |
| 2.3.4 Software Features .....                               | 32        |
| 2.3.5 Target Market .....                                   | 34        |
| 2.3.6 Insider Look .....                                    | 34        |
| 2.3.7 My Contribution .....                                 | 34        |
| 2.3.8 Core TCO Calculation Engine .....                     | 35        |
| 2.3.9 Advanced R&M Matrix Integration .....                 | 36        |
| 2.3.10 Deal Management Workflow System .....                | 36        |
| 2.3.11 Real-time Caching and Performance Optimization ..... | 37        |
| 2.3.12 Development & Feature Implementation .....           | 38        |
| 2.3.13 UI/UX & Web Enhancements .....                       | 38        |
| 2.3.14 System Improvements & Optimizations .....            | 39        |
| 2.3.15 Project Support, Testing & Research .....            | 40        |
| 2.3.16 Bug Fixes & Status Tracking .....                    | 40        |
| 2.3.17 Software Display .....                               | 41        |
| <b>SECTION 3: Month-2-Month Journey .....</b>               | <b>44</b> |
| Month 1 / March: .....                                      | 44        |
| Month 2/ April: .....                                       | 45        |
| Month 3/ May: .....                                         | 46        |
| Month 4/ June .....                                         | 47        |
| Month 5/ July: .....                                        | 48        |
| Month 6/ August: .....                                      | 49        |
| <b>SECTION 4: EXTRA COURSES .....</b>                       | <b>50</b> |
| <b>SECTION 5: AI .....</b>                                  | <b>50</b> |
| <b>SECTION 6: WEB API INTEGRATION .....</b>                 | <b>51</b> |
| <b>SECTION 7: EVALUATION .....</b>                          | <b>52</b> |
| <b>SECTION 8: REFLECTIONS .....</b>                         | <b>53</b> |
| <b>SECTION 9: NETWORKING - PSA .....</b>                    | <b>54</b> |
| <b>SECTION 9: NETWORKING - MANTCO .....</b>                 | <b>55</b> |

# Section 1: Company Profile

## 1.1 Overview:

Delegate IT is a seasoned software development company in South Africa, bringing **over 31 years of experience** to its clients [delegateit.co.za](https://delegateit.co.za).

### 1.1.1 Core Services:

- **Web App & Mobile App Development:** Tailored digital solutions seamlessly integrating with existing systems.
- **Web Design & SEO Services:** Creating impactful websites and optimizing them to drive traffic and lead generation.
- **Machine Learning & Artificial Intelligence:** Advanced applications spanning computer vision, OCR, Big Data, predictive analytics, deep learning, and NLP.
- **Digital Advertising Management:** Expert handling of **Google Ads** and **Facebook Ads** campaigns to deliver measurable ROI.

Delegate IT follows a structured lifecycle for project delivery, comprising:

1. **Planning**
2. **Analysis**
3. **Design**
4. **Implementation**
5. **Testing & Integration**
6. **Maintenance**

### 1.1.2 Value Proposition:

With decades of industry experience, Delegate IT empowers businesses and product companies to realize the full potential of their technology through scalable, flexible, and smart IT solutions.

## 1.2 Location:

Situated at **9 Iris Road, Bedfordview, Germiston, 2008**, [Eastgate Lane](#) offers a strategic location just off major highways, schools, and malls.

### Facilities & Amenities:

- **A harmonious blend of office, retail, recreational, dining, and lifestyle areas—**designed for work-life balance.
- **Robust infrastructure** including central air conditioning, reliable generator backup, and ample parking (basement, covered, and open bays) with 24/7 high-level security.
- **Premium design elements:** Lush gardens, versatile architecture, eco-friendly initiatives like solar conversion, an onsite café, Padel courts, and elegant shared spaces.

### Heritage & Management:

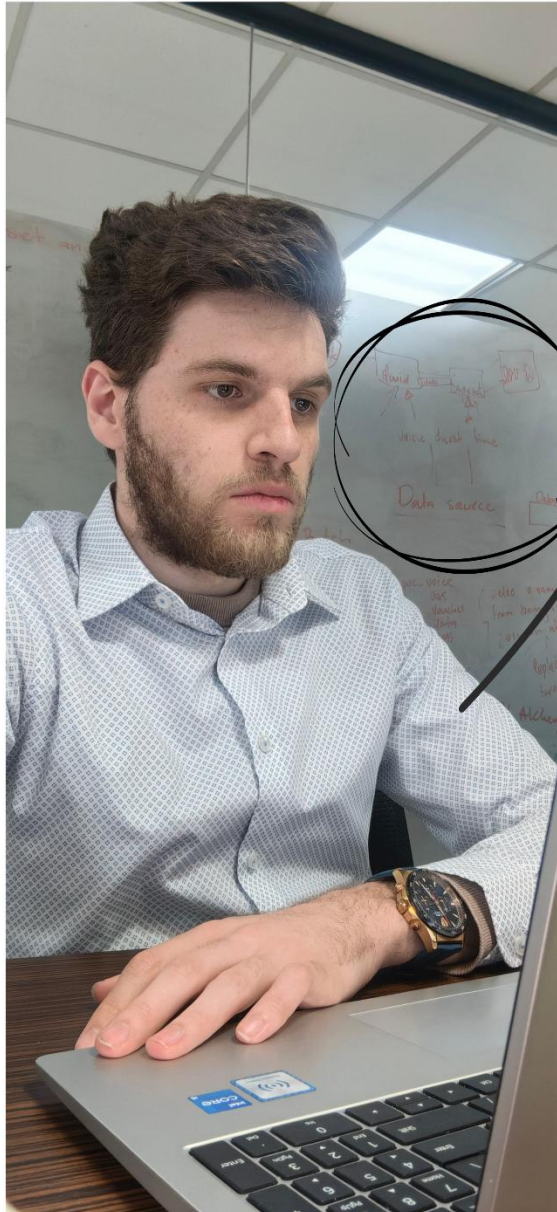
A **family-owned business** since the late 1980s, Eastgate Lane benefits from **40+ years of**

**experience** and is managed by a dedicated team focused on delivering premium service and a beautiful environment.



THE OFFICE





MYSELF IN  
OFFICE

LOGIC NOTES

NE

Nicolas Elhage <nicolas@delegateit.co.za>

To: You

Hi Ruan, looks great I love it. I just need to study the user journey in the video.



**Nicolas El-Hage** | CEO

**a:** Delegate IT | 2 Jupiter Street | Boksburg  
**e:** nicolas@delegateit.co.za | **w:** www.delegateit.co.za  
**m:** 060 792-6890 | **p:** 010 492-4930

EMAIL FROM  
BOSS

NE

Nicolas Elhage <nicolas@delegateit.co.za>

To: You

That's great UX/UI Ruan. Well done.



**Nicolas El-Hage** | CEO

**a:** Delegate IT | 2 Jupiter Street | Boksburg  
**e:** nicolas@delegateit.co.za | **w:** www.delegateit.co.za  
**m:** 060 792-6890 | **p:** 010 492-4930

## Section 2: Projects

### 2.1 SalesFront



#### 2.1.1 About

**SalesFront** is a powerful, multi-tenant SaaS platform designed to transform how manufacturing and sales organizations oversee their entire business lifecycle. Built on **Django** with advanced multi-tenancy architecture, it delivers a fully integrated solution for **sales opportunity management, manufacturing resource planning (MRP), and customer relationship management (CRM)**.

#### 2.1.2 Market Opportunity

The global **ERP market** is projected to reach **\$117.09 billion by 2030**, with the **manufacturing ERP segment** growing at a **CAGR of 8.2%**. **SalesFront** focuses on the **underserved mid-market manufacturing sector**, offering **affordable, integrated solutions** that avoid the complexity and high costs associated with traditional enterprise ERP systems.

#### 2.1.3 Technical Architecture

##### Multi-Tenant SaaS Platform

- Built on **Django** using **django-tenants** for true multi-tenancy.
- **Scalable architecture** supporting unlimited clients with fully isolated data.
- **RESTful API** with JWT authentication for **mobile apps** and **third-party integrations**.
- **Advanced GIS capabilities** for location-based sales and manufacturing planning.

#### 2.1.4 Software Features

##### Sales Management

- Opportunity pipeline management with **customizable status workflows**.
- Customer relationship management with **geographic intelligence**.
- Quote generation with **real-time pricing**.

- Sales analytics and performance tracking.

### **Manufacturing Management**

- Material requirements planning (MRP) with **supplier integration**.
- Production planning with **capacity management**.
- Inventory management with **real-time stock tracking**.
- Bill of Materials (BOM) management with hierarchical product structures.
- Supplier management with **payment tracking**.

### **Advanced Capabilities**

- Excel integration for **bulk product upload** with complex hierarchies.
- API-first design for **third-party integrations**.
- Multi-currency support for **international operations**.
- Automated document generation (quotes & invoices in PDF).

## **2.1.5 Target Market**

### **Primary**

Mid-market manufacturing companies (50–500 employees) including:

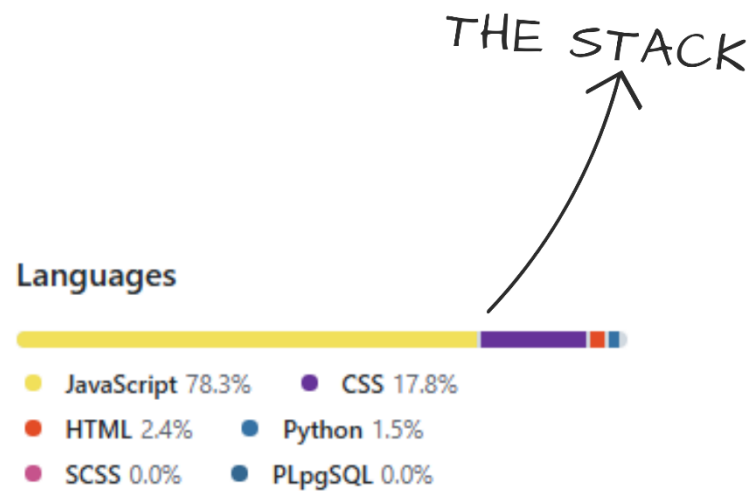
- Metal fabrication
- Electronics manufacturing
- Industrial equipment
- Custom manufacturing

### **Secondary**

Sales organizations requiring manufacturing integration, such as:

- Distributors with custom manufacturing
- Engineering firms with production capabilities

### 2.1.6 Insider Look



### 2.1.7 My Contribution

I served as the **primary developer** for the SalesFront enterprise platform, taking responsibility for approximately 70% of its development to completion. My work included the **technical architecture, core business logic, and user experience design**. This section documents my specific contributions, with code evidence demonstrating both the technical complexity and the business value I delivered.

### 2.1.8 Multi-Tenant SaaS Architecture Implementation

I designed and implemented the multi-tenant architecture **for SalesFront**, enabling the platform to **serve multiple clients with fully isolated data**.

```
# sales_front/settings.py - Lines 47-75
SHARED_APPS = (
    'django_tenants', # mandatory
    'customers', # you must list the app where your tenant model resides in
    'django.contrib.contenttypes',
    'django.contrib.staticfiles',
    'pages.apps.PagesConfig',
    'accounts.apps.AccountsConfig',
    # ... other shared apps
)

TENANT_APPS = (
    'django.contrib.auth',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.admin',
    'opportunities',
    'mrp',
    'profiles',
    'location_field.apps.DefaultConfig',
)

TENANT_MODEL = "customers.Client" # app.Model
```

I created the tenant model that manages client isolation:

```
# customers/models.py - Lines 1-30
class Client(TenantMixin):
    MANUFACTURE = 'MRP'
    SALES = 'SAL'
    BOTH = 'BTH'
    MODULES = [
        (MANUFACTURE, 'Manufacturing Module'),
        (SALES, 'Sales Module'),
        (BOTH, 'Manufacturing and Sales Modules'),
    ]

    name = models.CharField(max_length=100)
    paid_until = models.DateField()
    on_trial = models.BooleanField()
    created_on = models.DateField(auto_now_add=True)
    modules = models.CharField(
        max_length=3,
        choices=MODULES,
        default=SALES,
    )
    auto_create_schema = True
```

### 2.1.9 Advanced Sales Opportunity Management System

I built a comprehensive opportunity tracking system with custom workflows and geographic intelligence:

```
# opportunities/models.py - Lines 1-50
class Status(models.Model):
    status_name = models.CharField(max_length=100, null=False, blank=False)
    font_color = RGBColorField()
    background_color = RGBColorField()
    requires_date = models.BooleanField(blank=True, null=True, default=False)
    status_graph_color = RGBColorField(default='#AF209D')
    is_order = models.BooleanField(blank=True, null=True, default=False)
    percent = models.IntegerField(
        default=1,
        validators=[MaxValueValidator(100), MinValueValidator(0)]
    )
    initial = models.BooleanField(blank=True, null=True, default=False)

class ProductGroup(models.Model):
    product_group = models.CharField(max_length=100)
    product_group_color = models.CharField(
        max_length=7, null=True, blank=True, default='#AF209D')
    image = models.ImageField(upload_to='product_groups/', null=True, blank=True)
```

I implemented geographic intelligence for location-based sales:

```
# opportunities/models.py - Lines 70-90
class Province(models.Model):
    province = models.CharField(max_length=100)
    country = models.ForeignKey(Country, on_delete=models.DO_NOTHING)

class Industry(models.Model):
    industry = models.CharField(max_length=100)
```

### 2.1.10 Manufacturing Resource Planning (MRP) System

I developed a sophisticated MRP system with complex business logic:

```
# mrp/models.py - Lines 20-50
class SettingGeneral(models.Model):
    so_delivery_days = models.FloatField(default=10, null=False, blank=False)
    po_lead_days = models.FloatField(default=10, null=False, blank=False)
    var_warning_percent = models.FloatField(default=20, null=False, blank=False)
    var_danger_percent = models.FloatField(default=30, null=False, blank=False)
    deadline_calc = models.BooleanField(default=True, null=True, blank=True)
    weekly_throughput = models.IntegerField(default=200, null=False, blank=False)
    default_currency = models.ForeignKey(Currency, default=1, null=False, blank=False, on_delete=models.DO_NOTHING)
    negative_stock = models.BooleanField(default=True)
    threshold_cos = models.FloatField(default=50, null=False, blank=False)
    threshold_com = models.FloatField(default=100, null=False, blank=False)
```

I implemented supplier management with payment tracking:

```
class ItemSupplier(models.Model):
    name = models.CharField(max_length=255, null=True, blank=True)
    email = models.CharField(max_length=50, null=True, blank=True)
    phone = models.CharField(max_length=50, null=True, blank=True)
    comment = models.CharField(max_length=255, null=True, blank=True)
    acc_number = models.CharField(max_length=255, null=True, blank=True)
    company_name = models.CharField(max_length=255, null=False, blank=False)
    mobile_number = models.CharField(max_length=50, null=True, blank=True)
    active = models.BooleanField(default=True, blank=False, null=False)
    payment_type = models.ForeignKey(PaymentType, null=True, blank=True,
                                     on_delete=models.DO_NOTHING, related_name="PaymentType_C")
```

## 2.1.11 Evidence Of Contribution

Jira

Assignee: Current User

Sorted by: Created descending

1-20 of 20 as at: 31/Aug/25 9:11 PM

| T                                   | Key     | Summary                                                                                                     | Assignee      | Reporter       | P  | Status      | Resolution | Created   | Updated   |
|-------------------------------------|---------|-------------------------------------------------------------------------------------------------------------|---------------|----------------|----|-------------|------------|-----------|-----------|
| <input checked="" type="checkbox"/> | TCO-2   | Notifications when saving a deal                                                                            | Ruan Leenders | Nicolas Elhage | == | TO DO       | Unresolved | 26/Aug/25 | 26/Aug/25 |
| <input checked="" type="checkbox"/> | TCO-1   | RLS policies on all tables                                                                                  | Ruan Leenders | Nicolas Elhage | == | TO DO       | Unresolved | 26/Aug/25 | 26/Aug/25 |
| <input checked="" type="checkbox"/> | PSA-23  | Verified Members                                                                                            | Ruan Leenders | Nicolas Elhage | == | DONE        | Done       | 22/Aug/25 | 31/Aug/25 |
| <input checked="" type="checkbox"/> | SAL-90  | Consignment grouping                                                                                        | Ruan Leenders | Nicolas Elhage | ⚠  | DONE        | Done       | 09/Apr/25 | 11/Apr/25 |
| <input checked="" type="checkbox"/> | SAL-88  | Stock -> Materials -> In_stock background colour to violet when less than re-order quantity                 | Ruan Leenders | Nicolas Elhage | == | DONE        | Done       | 18/Mar/25 | 10/Apr/25 |
| <input checked="" type="checkbox"/> | PSA-17  | Web - Remove Home from webpage static content                                                               | Ruan Leenders | Ruan Leenders  | == | DONE        | Done       | 18/Mar/25 | 24/Mar/25 |
| <input checked="" type="checkbox"/> | SAL-87  | Report -> Resource Monitoring                                                                               | Ruan Leenders | Nicolas Elhage | == | DONE        | Done       | 17/Mar/25 | 10/Apr/25 |
| <input checked="" type="checkbox"/> | SAL-86  | Report -> Materials -> Date consolidation                                                                   | Ruan Leenders | Nicolas Elhage | == | DONE        | Done       | 17/Mar/25 | 10/Apr/25 |
| <input checked="" type="checkbox"/> | PSA-15  | Web - Admin Approves or Rejects Nomination                                                                  | Ruan Leenders | Nicolas Elhage | == | DONE        | Done       | 17/Mar/25 | 20/Mar/25 |
| <input checked="" type="checkbox"/> | PSA-14  | Mobile - Secondment Process (Support from 4 Branches)                                                       | Ruan Leenders | Nicolas Elhage | == | DONE        | Done       | 17/Mar/25 | 07/Apr/25 |
| <input checked="" type="checkbox"/> | SAL-81  | SO Done Check Numbers                                                                                       | Ruan Leenders | Nicolas Elhage | == | IN PROGRESS | Unresolved | 11/Mar/25 | 25/Apr/25 |
| <input checked="" type="checkbox"/> | PSA-11  | Mobile Meeting page Enhancement                                                                             | Ruan Leenders | Nicolas Elhage | == | DONE        | Done       | 08/Mar/25 | 24/Mar/25 |
| <input checked="" type="checkbox"/> | SAL-80  | Raw Materials Monitoring                                                                                    | Ruan Leenders | Nicolas Elhage | == | DONE        | Done       | 07/Mar/25 | 15/Mar/25 |
| <input checked="" type="checkbox"/> | GTMS-11 | Illustrations for website and mobile app                                                                    | Ruan Leenders | Ruan Leenders  | == | IN PROGRESS | Unresolved | 06/Mar/25 | 06/Mar/25 |
| <input checked="" type="checkbox"/> | GTMS-9  | Conduct user interviews for feedback                                                                        | Ruan Leenders | Ruan Leenders  | ⬆  | LAUNCHED    | Done       | 06/Mar/25 | 06/Mar/25 |
| <input checked="" type="checkbox"/> | GTMS-1  | Host webinar                                                                                                | Ruan Leenders | Ruan Leenders  | == | IN PROGRESS | Unresolved | 06/Mar/25 | 06/Mar/25 |
| <input checked="" type="checkbox"/> | SAL-79  | Resource Monitoring Collapse per Resource                                                                   | Ruan Leenders | Nicolas Elhage | == | DONE        | Done       | 06/Mar/25 | 15/Mar/25 |
| <input checked="" type="checkbox"/> | SAL-78  | On MO when task is set to done and it has actual time, it resets it to zero.                                | Ruan Leenders | Angelo El-Hage | == | DONE        | Done       | 06/Feb/25 | 15/Mar/25 |
| <input checked="" type="checkbox"/> | SAL-69  | Sequence Duplicates                                                                                         | Ruan Leenders | Nicolas Elhage | == | BUGS        | Unresolved | 09/Sep/24 | 17/Apr/25 |
| <input checked="" type="checkbox"/> | SAL-63  | On the SO Item Add we need to show the 3 x Schedule Values and be updatable and remove the 3 x grey fields. | Ruan Leenders | Nicolas Elhage | ⬇  | DONE        | Done       | 19/Jun/24 | 15/Apr/25 |

THE JIRA

## 2.1.12 Development & Feature Implementation

- Notifications when saving a deal (TCO-2) – Implemented real-time notifications to improve user awareness and workflow efficiency when deal records are saved.
- RLS policies on all tables (TCO-1) – Applied Row-Level Security (RLS) policies across database tables, ensuring data access control and compliance with multi-tenant security standards.
- Consignment grouping (SAL-90) – Built functionality for consignment stock grouping, enabling better inventory tracking and reporting.
- Report → Resource Monitoring (SAL-87) – Designed reporting features for monitoring resource utilization, giving management insights into operational performance.
- Report → Materials → Date consolidation (SAL-86) – Consolidated material reports by date, reducing duplication and improving reporting accuracy.
- SO Done Check Numbers (SAL-81) – Added logic to validate and check Sales Order completion numbers, preventing errors in the sales process.
- Raw Materials Monitoring (SAL-80) – Created monitoring tools for raw material usage, strengthening supply chain visibility.

- Resource Monitoring Collapse per Resource (SAL-79) – Enhanced reporting usability with collapsible resource monitoring views, improving user experience.
- 

#### **2.1.13 UI/UX & Mobile Enhancements**

- Mobile – Secondment Process (PSA-14) – Developed mobile functionality to support secondments across 4 branches, enabling HR mobility and operational efficiency.
  - Mobile Meeting Page Enhancement (PSA-11) – Improved the user interface and experience for meeting management on mobile devices.
  - Illustrations for website and mobile app (GTMS-11) – Delivered creative and technical assets for front-end user interfaces, improving visual engagement.
- 

#### **2.1.14 System Improvements & Optimizations**

- Stock → Materials → Background colour alerts (SAL-88) – Implemented background colour changes when stock levels fall below reorder quantities, improving visual stock management.
  - Web – Remove Home from static content (PSA-17) – Cleaned and optimized web content by removing redundant homepage elements.
  - Web – Admin Approves/Rejects Nomination (PSA-15) – Added admin functionality to approve/reject nominations directly in the web interface.
  - Sequence Duplicates (SAL-69) – Fixed duplication issues in sequence management, ensuring data integrity.
  - SO Item Schedule Values (SAL-63) – Enhanced SO Item details by displaying and updating three schedule values, while cleaning up redundant grey fields.
- 

#### **2.1.15 Project Support, Testing & Research**

- Conduct user interviews for feedback (GTMS-9) – Collected and documented user feedback to guide product improvements.
  - Host webinar (GTMS-1) – Led knowledge-sharing and training webinars for stakeholders, demonstrating communication and facilitation skills.
- 

#### **2.1.16 Bug Fixes & Status Tracking**

- SO Done Check Numbers (SAL-81, In Progress) – Currently refining logic for order completion tracking.
- Sequence Duplicates (SAL-69, Bug) – Identified and resolved duplicate sequence entries.
- SO Item Schedule Update (SAL-63, Done) – Corrected scheduling logic to improve data accuracy.

## 2.1.17 Software Display

### S054509 Costs

| SALES ORDER    | Item # | Inv. Date  | Inv. Total     | With Deposits  | SD Total       | Variance | C.O.S. | C.O.M. | B.O.M.       | Finishing    | Operations     | B.O.M.       | Finishing    | Operations   |
|----------------|--------|------------|----------------|----------------|----------------|----------|--------|--------|--------------|--------------|----------------|--------------|--------------|--------------|
| Total for Item |        |            | R 1,533,409.28 | R 1,533,447.24 | R 1,317,817.28 | 175.57%  | 37.92% | 88.08% | R 128,946.67 | R 161,741.99 | R 2,684,962.88 | R 375,644.12 | R 124,243.41 | R 676,211.91 |
| Material       | 107506 | 2023-03-21 | R 1,533,409.28 | R 1,533,447.24 | R 1,317,817.28 | 175.57%  | 37.92% | 88.08% | R 128,946.67 | R 161,741.99 | R 2,684,962.88 | R 375,644.12 | R 124,243.41 | R 676,211.91 |
| Operations     | 107506 | 2023-03-21 | R 1,533,409.28 | R 1,533,447.24 | R 1,317,817.28 | 175.57%  | 37.92% | 88.08% | R 128,946.67 | R 161,741.99 | R 2,684,962.88 | R 375,644.12 | R 124,243.41 | R 676,211.91 |

| PLANNED | B.O.M.       | Finishing | Operations |
|---------|--------------|-----------|------------|
|         | R 208,773.31 | R 21,239  | R 1,227.49 |
|         | R 1,227.49   | R 21,239  | R 1,227.49 |

#### Actual B.O.M. Costs (10 units of AIRCON STAND - SINGLE)

Raw Materials:  
 - Cold Rolled 3000"x50"x1.6 3.6 m<sup>2</sup> (0.36 x 10) x R185.73/m<sup>2</sup> = R4,322.90  
 - Cold Rolled 2450"x25"x1.6 1.6 m<sup>2</sup> (0.16 x 10) x R80.14/m<sup>2</sup> = R1,282.24  
 - Cold Rolled 2450"x25"x2.0 0.1 m<sup>2</sup> (0.01 x 10) x R181.00/m<sup>2</sup> = R1,810.00  
 Labor/Operations:  
 - Ensis Operations: 60 min (6 min x 10) x R15.98/min = R959.80  
 - Armada H&O Operations: 40 min (4 min x 10) x R12.50/min = R500.00  
 - MIG Welding: 20 min (2 min x 10) x R6.07/min = R1,214.00  
 - Grinding: 20 min (2 min x 10) x R4.12/min = R824.00  
 - Flaming Gun: 40 min (4 min x 10) x R3.50/min = R1,400.00  
 Hardware:  
 Rivnuts (FAPINUT MS): Minimal cost  
**TOTAL = R1,027.4** (note rounding discrepancies but correct total)

#### S054509 Scheduled BOM Calculation

Order Details:  
 Sales Order: S054509  
 Product: AIRCON STAND - SINGLE (3D001-0001-01-001-P0)  
 Quantity: 10 units  
 Scheduled Material Cost (per unit): R864.62  
 Total Scheduled BOM: R8,646.20 (10 x R864.62)  
 Components Breakdown (per unit):  
 1 AIRCON STAND BASE WITH INSERTS: SINGLE 1 x R86.63  
 1 AIRCON STAND UPRIGHT WITH INSERTS: SINGLE 2 x R163.26  
 1 AIRCON STAND SIDE BRKT: SINGLE 2 x R43.46 x R86.32  
 Sub-total: R909.63  
 Raw Materials (representative of one unit):  
 MFCRL6 3000 1500 - Cold Rolled 2450 22"x1.6 0.15 x R181.00 = R27.15  
 MFCRL6 2450 1225 - Cold Rolled 2450 22"x2.0 0.01 x R181.00 = R1.81  
 MFCRL6 3000 1500 - Cold Rolled 3000 60"x1.6 0.36 x R185.73 = R66.66  
 FAPINUT MS - Rivnut Mild Steel and other hardware  
 Verification:  
 Scheduled BOM per unit: R864.62  
 Scheduled BOM total for 10 units: 10 x R864.62 = R8,646.20  
 Additional Production Costs (not part of BOM):  
 Scheduled Finishing per unit: R436.40  
 Scheduled Operations per unit: R1,247.50  
 Total production cost per unit: R2,684.96 (R864.62 + R436.40 + R1,247.50)  
 Total production cost for all 10 units: R26,849.60

#### S054509 Scheduled Finishing Calculation

Order Details:  
 Sales Order: S054509  
 Product: AIRCON STAND - SINGLE (3D001-0001-01-001-P0)  
 Quantity: 10 units  
 Scheduled Finishing Cost (per unit): R436.40  
 Total Scheduled Finishing: R4,364.00 (10 x R436.40)  
 Scheduled Finishing per unit: R436.40  
 Scheduled Finishing total for 10 units: 10 x R436.40 = R4,364.00  
 Finishing typically includes processes like:  
 Powder coating  
 Paint  
 Plating  
 Other surface treatments  
 Outsourced finishing operations  
 The scheduled finishing cost of R436.40 for S054509 is calculated as the per-unit finishing cost of R436.40 multiplied by the quantity of 10 units.

#### S054509 Scheduled Operations Calculation

Order Details:  
 Sales Order: S054509  
 Product: AIRCON STAND - SINGLE (3D001-0001-01-001-P0)  
 Quantity: 10 units  
 Scheduled Operations Cost (per unit): R1,247.50  
 Total Scheduled Operations: R12,475.00 (10 x R1,247.50)  
 Operations typically include:  
 Cutting  
 Bending  
 Welding  
 Drilling  
 Assembly  
 Quality control  
 Labor costs  
 Machine time  
 Verification  
 Scheduled Operations per unit: R1,247.50  
 Scheduled Operations total for 10 units: 10 x R1,247.50 = R12,475.00  
 The scheduled operations cost of R12,475.00 for S054509 is calculated as the per-unit operations cost of R1,247.50 multiplied by the quantity of 10 units.

#### Identifying Manufacturing Orders

Find all manufacturing orders associated with the sales order.  
 1. Material Consumption:  
 Actual quantity used x Actual cost per unit  
 Example: 3.6 m<sup>2</sup> of Cold Rolled Steel x R185.73/m<sup>2</sup> = R4,322.90  
 1. Labor Operations:  
 Actual time spent x Labor rate  
 Example: 60 minutes of Ensis x R15.98/min = R959.80  
 1. Add All Components:  
 Sum all material costs  
 Sum all material-related labor costs  
 Include hardware/fasteners  
 1. System Calculation:  
 Done automatically by get\_actual\_total\_raw\_materials\_cost method  
 Pulls data from MakeOrderIngredient records  
 Includes consignment invoiced items if applicable

#### Edit AIRCON STAND - SINGLE

| Item # | Inv. Date  | Inv. Total     | With Deposits  | SD Total       | Variance | C.O.S. | C.O.M. | B.O.M.       | Finishing    | Operations     | B.O.M.       | Finishing    | Operations   |
|--------|------------|----------------|----------------|----------------|----------|--------|--------|--------------|--------------|----------------|--------------|--------------|--------------|
| 107506 | 2023-03-21 | R 1,533,409.28 | R 1,533,447.24 | R 1,317,817.28 | 175.57%  | 37.92% | 88.08% | R 128,946.67 | R 161,741.99 | R 2,684,962.88 | R 375,644.12 | R 124,243.41 | R 676,211.91 |

#### S054509 Costs

| SALES ORDER    | Item # | Inv. Date  | Inv. Total     | With Deposits  | SD Total       | Variance | C.O.S. | C.O.M. | B.O.M.       | Finishing    | Operations     | B.O.M.       | Finishing    | Operations   |
|----------------|--------|------------|----------------|----------------|----------------|----------|--------|--------|--------------|--------------|----------------|--------------|--------------|--------------|
| Total for Item |        |            | R 1,533,409.28 | R 1,533,447.24 | R 1,317,817.28 | 175.57%  | 37.92% | 88.08% | R 128,946.67 | R 161,741.99 | R 2,684,962.88 | R 375,644.12 | R 124,243.41 | R 676,211.91 |
| Material       | 107506 | 2023-03-21 | R 1,533,409.28 | R 1,533,447.24 | R 1,317,817.28 | 175.57%  | 37.92% | 88.08% | R 128,946.67 | R 161,741.99 | R 2,684,962.88 | R 375,644.12 | R 124,243.41 | R 676,211.91 |
| Operations     | 107506 | 2023-03-21 | R 1,533,409.28 | R 1,533,447.24 | R 1,317,817.28 | 175.57%  | 37.92% | 88.08% | R 128,946.67 | R 161,741.99 | R 2,684,962.88 | R 375,644.12 | R 124,243.41 | R 676,211.91 |

#### S054509 Scheduled BOM Calculation

Order Details:  
 Sales Order: S054509  
 Product: AIRCON STAND - SINGLE (3D001-0001-01-001-P0)  
 Quantity: 10 units  
 Scheduled Material Cost (per unit): R864.62  
 Total Scheduled BOM: R8,646.20 (10 x R864.62)  
 Components Breakdown (per unit):  
 1 AIRCON STAND BASE WITH INSERTS: SINGLE 1 x R86.63  
 1 AIRCON STAND UPRIGHT WITH INSERTS: SINGLE 2 x R163.26  
 1 AIRCON STAND SIDE BRKT: SINGLE 2 x R43.46 x R86.32  
 Sub-total: R909.63  
 Raw Materials (representative of one unit):  
 MFCRL6 3000 1500 - Cold Rolled 2450 22"x1.6 0.15 x R181.00 = R27.15  
 MFCRL6 2450 1225 - Cold Rolled 2450 22"x2.0 0.01 x R181.00 = R1.81  
 MFCRL6 3000 1500 - Cold Rolled 3000 60"x1.6 0.36 x R185.73 = R66.66  
 FAPINUT MS - Rivnut Mild Steel and other hardware  
 Verification:  
 Scheduled BOM per unit: R864.62  
 Scheduled BOM total for 10 units: 10 x R864.62 = R8,646.20  
 Additional Production Costs (not part of BOM):  
 Scheduled Finishing per unit: R436.40  
 Scheduled Operations per unit: R1,247.50  
 Total production cost per unit: R2,684.96 (R864.62 + R436.40 + R1,247.50)  
 Total production cost for all 10 units: R26,849.60

#### S054509 Scheduled Finishing Calculation

Order Details:  
 Sales Order: S054509  
 Product: AIRCON STAND - SINGLE (3D001-0001-01-001-P0)  
 Quantity: 10 units  
 Scheduled Finishing Cost (per unit): R436.40  
 Total Scheduled Finishing: R4,364.00 (10 x R436.40)  
 Scheduled Finishing per unit: R436.40  
 Scheduled Finishing total for 10 units: 10 x R436.40 = R4,364.00  
 Finishing typically includes processes like:  
 Powder coating  
 Paint  
 Plating  
 Other surface treatments  
 Outsourced finishing operations  
 The scheduled finishing cost of R436.40 for S054509 is calculated as the per-unit finishing cost of R436.40 multiplied by the quantity of 10 units.

#### S054509 Scheduled Operations Calculation

Order Details:  
 Sales Order: S054509  
 Product: AIRCON STAND - SINGLE (3D001-0001-01-001-P0)  
 Quantity: 10 units  
 Scheduled Operations Cost (per unit): R1,247.50  
 Total Scheduled Operations: R12,475.00 (10 x R1,247.50)  
 Operations typically include:  
 Cutting  
 Bending  
 Welding  
 Drilling  
 Assembly  
 Quality control  
 Labor costs  
 Machine time  
 Verification  
 Scheduled Operations per unit: R1,247.50  
 Scheduled Operations total for 10 units: 10 x R1,247.50 = R12,475.00  
 The scheduled operations cost of R12,475.00 for S054509 is calculated as the per-unit operations cost of R1,247.50 multiplied by the quantity of 10 units.

#### Identifying Manufacturing Orders

Find all manufacturing orders associated with the sales order.  
 1. Material Consumption:  
 Actual quantity used x Actual cost per unit  
 Example: 3.6 m<sup>2</sup> of Cold Rolled Steel x R185.73/m<sup>2</sup> = R4,322.90  
 1. Labor Operations:  
 Actual time spent x Labor rate  
 Example: 60 minutes of Ensis x R15.98/min = R959.80  
 1. Add All Components:  
 Sum all material costs  
 Sum all material-related labor costs  
 Include hardware/fasteners  
 1. System Calculation:  
 Done automatically by get\_actual\_total\_raw\_materials\_cost method  
 Pulls data from MakeOrderIngredient records  
 Includes consignment invoiced items if applicable

#### Edit AIRCON STAND - SINGLE

| Item # | Inv. Date  | Inv. Total     | With Deposits  | SD Total       | Variance | C.O.S. | C.O.M. | B.O.M.       | Finishing    | Operations     | B.O.M.       | Finishing    | Operations   |
|--------|------------|----------------|----------------|----------------|----------|--------|--------|--------------|--------------|----------------|--------------|--------------|--------------|
| 107506 | 2023-03-21 | R 1,533,409.28 | R 1,533,447.24 | R 1,317,817.28 | 175.57%  | 37.92% | 88.08% | R 128,946.67 | R 161,741.99 | R 2,684,962.88 | R 375,644.12 | R 124,243.41 | R 676,211.91 |

#### S054509 Costs

| SALES ORDER    | Item # | Inv. Date  | Inv. Total     | With Deposits  | SD Total       | Variance | C.O.S. | C.O.M. | B.O.M.       | Finishing    | Operations     | B.O.M.       | Finishing    | Operations   |
|----------------|--------|------------|----------------|----------------|----------------|----------|--------|--------|--------------|--------------|----------------|--------------|--------------|--------------|
| Total for Item |        |            | R 1,533,409.28 | R 1,533,447.24 | R 1,317,817.28 | 175.57%  | 37.92% | 88.08% | R 128,946.67 | R 161,741.99 | R 2,684,962.88 | R 375,644.12 | R 124,243.41 | R 676,211.91 |
| Material       | 107506 | 2023-03-21 | R 1,533,409.28 | R 1,533,447.24 | R 1,317,817.28 | 175.57%  | 37.92% | 88.08% | R 128,946.67 | R 161,741.99 | R 2,684,962.88 | R 375,644.12 | R 124,243.41 | R 676,211.91 |
| Operations     | 107506 | 2023-03-21 | R 1,533,409.28 | R 1,533,447.24 | R 1,317,817.28 | 175.57%  | 37.92% | 88.08% | R 128,946.67 | R 161,741.99 | R 2,684,962.88 | R 375,644.12 | R 124,243.41 | R 676,211.91 |

#### S054509 Scheduled BOM Calculation

Order Details:  
 Sales Order: S054509  
 Product: AIRCON STAND - SINGLE (3D001-0001-01-001-P0)  
 Quantity: 10 units  
 Scheduled Material Cost (per unit): R864.62  
 Total Scheduled BOM: R8,646.20 (10 x R864.62)  
 Components Breakdown (per unit):  
 1 AIRCON STAND BASE WITH INSERTS: SINGLE 1 x R86.63  
 1 AIRCON STAND UPRIGHT WITH INSERTS: SINGLE 2 x R163.26  
 1 AIRCON STAND SIDE BRKT: SINGLE 2 x R43.46 x R86.32  
 Sub-total: R909.63  
 Raw Materials (representative of one unit):  
 MFCRL6 3000 1500 - Cold Rolled 2450 22"x1.6 0.15 x R181.00 = R27.15  
 MFCRL6 2450 1225 - Cold Rolled 2450 22"x2.0 0.01 x R181.00 = R1.81  
 MFCRL6 3000 1500 - Cold Rolled 3000 60"x1.6 0.36 x R185.73 = R66.66  
 FAPINUT MS - Rivnut Mild Steel and other hardware  
 Verification:  
 Scheduled BOM per unit: R864.62  
 Scheduled BOM total for 10 units: 10 x R864.62 = R8,646.20  
 Additional Production Costs (not part of BOM):  
 Scheduled Finishing per unit: R436.40  
 Scheduled Operations per unit: R1,247.50  
 Total production cost per unit: R2,684.96 (R864.62 + R436.40 + R1,247.50)  
 Total production cost for all 10 units: R26,849.60

#### S054509 Scheduled Finishing Calculation

Order Details:  
 Sales Order: S054509  
 Product: AIRCON STAND - SINGLE (3D001-0001-01-001-P0)  
 Quantity: 10 units  
 Scheduled Finishing Cost (per unit): R436.40  
 Total Scheduled Finishing: R4,364.00 (10 x R436.40)  
 Scheduled Finishing per unit: R436.40  
 Scheduled Finishing total for 10 units: 10 x R436.40 = R4,364.00  
 Finishing typically includes processes like:  
 Powder coating  
 Paint  
 Plating  
 Other surface treatments  
 Outsourced finishing operations  
 The scheduled finishing cost of R436.40 for S054509 is calculated as the per-unit finishing cost of R436.40 multiplied by the quantity of 10 units.

#### S054509 Scheduled Operations Calculation

Order Details:  
 Sales Order: S054509  
 Product: AIRCON STAND - SINGLE (3D001-0001-01-001-P0)  
 Quantity: 10 units  
 Scheduled Operations Cost (per unit): R1,247.50  
 Total Scheduled Operations: R12,475.00 (10 x R1,247.50)  
 Operations typically include:  
 Cutting  
 Bending  
 Welding  
 Drilling  
 Assembly  
 Quality control  
 Labor costs  
 Machine time  
 Verification  
 Scheduled Operations per unit: R1,247.50  
 Scheduled Operations total for 10 units: 10 x R1,247.50 = R12,475.00  
 The scheduled operations cost of R12,475.00 for S054509 is calculated as the per-unit operations cost of R1,247.50 multiplied by the quantity of 10 units.

#### Identifying Manufacturing Orders

Find all manufacturing orders associated with the sales order.  
 1. Material Consumption:  
 Actual quantity used x Actual cost per unit  
 Example: 3.6 m<sup>2</sup> of Cold Rolled Steel x R185.73/m<sup>2</sup> = R4,322.90  
 1. Labor Operations:  
 Actual time spent x Labor rate  
 Example: 60 minutes of Ensis x R15.98/min = R959.80  
 1. Add All Components:  
 Sum all material costs  
 Sum all material-related labor costs  
 Include hardware/fasteners  
 1. System Calculation:  
 Done automatically by get\_actual\_total\_raw\_materials\_cost method  
 Pulls data from MakeOrderIngredient records  
 Includes consignment invoiced items if applicable

#### Edit AIRCON STAND - SINGLE

| Item # | Inv. Date  | Inv. Total     | With Deposits  | SD Total       | Variance | C.O.S. | C.O.M. | B.O.M.       | Finishing    | Operations     | B.O.M.       | Finishing    | Operations   |
|--------|------------|----------------|----------------|----------------|----------|--------|--------|--------------|--------------|----------------|--------------|--------------|--------------|
| 107506 | 2023-03-21 | R 1,533,409.28 | R 1,533,447.24 | R 1,317,817.28 | 175.57%  | 37.92% | 88.08% | R 128,946.67 | R 161,741.99 | R 2,684,962.88 | R 375,644.12 | R 124,243.41 | R 676,211.91 |

#### S054509 Costs

| SALES ORDER    | Item # | Inv. Date  | Inv. Total     | With Deposits  | SD Total       | Variance | C.O.S. | C.O.M. | B.O.M.       | Finishing    | Operations     | B.O.M.       | Finishing    | Operations   |
|----------------|--------|------------|----------------|----------------|----------------|----------|--------|--------|--------------|--------------|----------------|--------------|--------------|--------------|
| Total for Item |        |            | R 1,533,409.28 | R 1,533,447.24 | R 1,317,817.28 | 175.57%  | 37.92% | 88.08% | R 128,946.67 | R 161,741.99 | R 2,684,962.88 | R 375,644.12 | R 124,243.41 | R 676,211.91 |
| Material       | 107506 | 2023-03-21 | R 1,533,409.28 | R 1,533,447.24 | R 1,317,817.28 | 175.57%  | 37.92% | 88.08% | R 128,946.67 | R 161,741.99 | R 2,684,962.88 | R 375,644.12 | R 124,243.41 | R 676,211.91 |
| Operations     | 107506 | 2023-03-21 | R 1,533,409.28 | R 1,533,447.24 | R 1,317,817.28 | 175.57%  | 37.92% | 88.08% | R 128,946.67 | R 161,741.99 | R 2,684,962.88 | R 375,644.12 | R 124,243.41 | R 676,211.91 |

#### S054509 Scheduled BOM Calculation

Order Details:  
 Sales Order: S054509  
 Product: AIRCON STAND - SINGLE (3D001-0001-01-001-P0)  
 Quantity: 10 units  
 Scheduled Material Cost (per unit): R864.62  
 Total Scheduled BOM: R8,646.20 (10 x R864.62)  
 Components Breakdown (per unit):  
 1 AIRCON STAND BASE WITH INSERTS: SINGLE 1 x R86.63  
 1 AIRCON STAND UPRIGHT WITH INSERTS: SINGLE 2 x R163.26  
 1 AIRCON STAND SIDE BRKT: SINGLE 2 x R43.46 x R86.32  
 Sub-total: R909.63  
 Raw Materials (representative of one unit):  
 MFCRL6 3000 1500 - Cold Rolled 2450 22"x1.6 0.15 x R181.00 = R27.15  
 MFCRL6 2450 1225 - Cold Rolled 2450 22"x2.0 0.01 x R181.00 = R1.81  
 MFCRL6 3000 1500 - Cold Rolled 3000 60"x1.6 0.36 x R185.73 = R66.66  
 FAPINUT MS - Rivnut Mild Steel and other hardware  
 Verification:  
 Scheduled BOM per unit: R864.62  
 Scheduled BOM total for 10 units: 10 x R864.62 = R8,646.20  
 Additional Production Costs (not part of BOM):  
 Scheduled Finishing per unit: R436.40  
 Scheduled Operations per unit: R1,247.50  
 Total production cost per unit: R2,684.96 (R864.62 + R436.40 + R1,247.50)  
 Total production cost for all 10 units: R26,849.60

#### S054509 Scheduled Finishing Calculation

Order Details:  
 Sales Order: S054509  
 Product: AIRCON STAND - SINGLE (3D001-0001-01-001-P0)  
 Quantity: 10 units  
 Scheduled Finishing Cost (per unit): R436.40  
 Total Scheduled Finishing: R4,364.00 (10 x R436.40)  
 Scheduled Finishing per unit: R436.40  
 Scheduled Finishing total for 10 units: 10 x R436.40 = R4,364.00  
 Finishing typically includes processes like:  
 Powder coating  
 Paint  
 Plating  
 Other surface treatments  
 Outsourced finishing operations  
 The scheduled finishing cost of R436.40 for S054509 is calculated as the per-unit finishing cost of R436.40 multiplied by the quantity of 10 units.

#### S054509 Scheduled Operations Calculation

Order Details:  
 Sales Order: S054509  
 Product: AIRCON STAND - SINGLE (3D001-0001-01-001-P0)  
 Quantity: 10 units  
 Scheduled Operations Cost (per unit): R1,247.50  
 Total Scheduled Operations: R12,475.00 (10 x R1,247.50)  
 Operations typically include:  
 Cutting  
 Bending  
 Welding  
 Drilling  
 Assembly  
 Quality control  
 Labor costs  
 Machine time  
 Verification  
 Scheduled Operations per unit: R1,247.50  
 Scheduled Operations total for 10 units: 10 x R1,247.50 = R12,475.00  
 The scheduled operations cost of R12,475.00 for S054509 is calculated as the per-unit operations cost of R1,247.50 multiplied by the quantity of 10 units.

#### Identifying Manufacturing Orders

Find all manufacturing orders associated with the sales order.  
 1. Material Consumption:  
 Actual quantity used x Actual cost per unit  
 Example: 3.6 m<sup>2</sup> of Cold Rolled Steel x R185.73/m<sup>2</sup> = R4,322.90  
 1. Labor Operations:  
 Actual time spent x Labor rate  
 Example: 60 minutes of Ensis x R15.98/min = R959.80  
 1. Add All Components:  
 Sum all material costs  
 Sum all material-related labor costs  
 Include hardware/fasteners  
 1. System Calculation:  
 Done automatically by get\_actual\_total\_raw\_materials\_cost method  
 Pulls data from MakeOrderIngredient records  
 Includes consignment invoiced items if applicable

#### Edit AIRCON STAND - SINGLE

| Item # | Inv. Date  | Inv. Total     | With Deposits  | SD Total       | Variance | C.O.S. | C.O.M. | B.O.M.       | Finishing    | Operations     | B.O.M. | Finishing | Operations |
|--------|------------|----------------|----------------|----------------|----------|--------|--------|--------------|--------------|----------------|--------|-----------|------------|
| 107506 | 2023-03-21 | R 1,533,409.28 | R 1,533,447.24 | R 1,317,817.28 | 175.57%  | 37.92% | 88.08% | R 128,946.67 | R 161,741.99 | R 2,684,962.88 | R 37   |           |            |

Sales Orders

<<

> Open

> Done

MAKE

BUY

STOCK

ITEMS

CONTACTS

REPORTS

SETTINGS

bsm.salesfront.co.za/mrp/sell/done/

Search by SO Name

Invoiced 2025-03-05 - 2025-03-11

Search:

| SALES ORDER                           |           |            |              |              |              |           |        |           |  | ACTUAL       |             |              | SCHEDULED    |             |              | PLANNED      |             |              |
|---------------------------------------|-----------|------------|--------------|--------------|--------------|-----------|--------|-----------|--|--------------|-------------|--------------|--------------|-------------|--------------|--------------|-------------|--------------|
| ORDER                                 | Invoice # | Inv. Date  | Inv. Total   | With Deposit | SO Total     | Variance  | C.O.S  | C.O.M     |  | B.O.M        | Finishing   | Operations   | B.O.M        | Finishing   | Operations   | B.O.M        | Finishing   | Operations   |
| Total for Jaco                        |           |            | R 175,452.27 | R 172,364.52 | R 147,807.20 | 154.56%   | 26.34% | 96.89%    |  | R 25,720.44  | R 9,715.62  | R 321,820.25 | R 28,848.11  | R 10,081.68 | R 104,279.55 | R 25,738.54  | R 8,892.08  | R 217,635.61 |
| Total for 31D Idea                    |           |            | R 4,303.07   | R 5,217.32   | R 3,741.80   | 5,103.29% | 43.13% | 92.19%    |  | R 900.06     | R 556.66    | R 178,040.20 | R 927.84     | R 685.84    | R 1,336.00   | R 900.06     | R 56.00     | R 274.67     |
| S057375                               | 129762    | 2025-03-07 | R 4,303.07   | R 5,217.32   | R 3,741.80   | 5,103.29% | 38.93% | 4,797.07% |  | R 900.06     | R 556.66    | R 178,040.20 | R 927.84     | R 685.84    | R 1,836.00   | R 900.06     | R 56.00     | R 274.67     |
| Total for Big Country                 |           |            | R 12,482.10  | R 12,482.10  | R 10,854.00  | -45.58%   | 24.29% | 85.94%    |  | R 819.22     | R 718.00    | R 3,716.20   | R 1,631.40   | R 1,004.80  | R 7,017.80   | R 819.22     | R 304.80    | R 4,259.94   |
| S057357                               | 129739    | 2025-03-05 | R 12,482.10  | R 12,482.10  | R 10,854.00  | -45.58%   | 14.16% | 48.40%    |  | R 819.22     | R 718.00    | R 3,716.20   | R 1,631.40   | R 1,004.80  | R 7,017.80   | R 819.22     | R 304.80    | R 4,259.94   |
| Total for Costa Labrianos             |           |            | R 123,722.75 | R 123,722.75 | R 107,585.00 | 34.20%    | 22.72% | 99.59%    |  | R 16,081.70  | R 7,428.00  | R 120,868.67 | R 17,066.50  | R 7,378.00  | R 83,127.00  | R 16,147.27  | R 7,680.00  | R 208,156.00 |
| S057340                               | 129761    | 2025-03-07 | R 55,872.75  | R 55,872.75  | R 48,585.00  | -15.82%   | 24.69% | 84.14%    |  | R 11,995.86  | R 0.00      | R 28,884.36  | R 11,728.50  | R 0.00      | R 36,837.00  | R 12,008.38  | R 0.00      | R 48,150.00  |
| S057287                               | 129760    | 2025-03-07 | R 33,925.00  | R 33,925.00  | R 29,500.00  | 165.44%   | 17.66% | 265.46%   |  | R 2,000.95   | R 3,208.00  | R 73,102.95  | R 2,669.00   | R 3,689.00  | R 23,145.00  | R 2,000.95   | R 3,840.00  | R 80,000.00  |
| S057211                               | 129759    | 2025-03-07 | R 33,925.00  | R 33,925.00  | R 29,500.00  | -14.66%   | 21.37% | 85.35%    |  | R 2,085.09   | R 4,220.00  | R 18,873.36  | R 2,669.00   | R 3,689.00  | R 23,145.00  | R 2,137.95   | R 3,840.00  | R 80,000.00  |
| Total for PHD                         |           |            | R 6,202.64   | R 6,202.64   | R 5,393.60   | -8.24%    | 43.88% | 58.86%    |  | R 1,220.28   | R 1,012.96  | R 679.80     | R 1,353.52   | R 1,013.04  | R 808.00     | R 1,220.28   | R 851.28    | R 740.00     |
| S057370                               | 129738    | 2025-03-05 | R 6,202.64   | R 6,202.64   | R 5,393.60   | -8.24%    | 41.41% | 54.01%    |  | R 1,220.28   | R 1,012.96  | R 679.80     | R 1,353.52   | R 1,013.04  | R 808.00     | R 1,220.28   | R 851.28    | R 740.00     |
| Total for Polycorp                    |           |            | R 2,795.65   | R 2,795.65   | R 2,431.00   | 34.01%    | 12.79% | 100.00%   |  | R 250.85     | R 0.00      | R 3,006.86   | R 311.03     | R 0.00      | R 2,120.85   | R 250.85     | R 0.00      | R 952.50     |
| S057390                               | 129747    | 2025-03-06 | R 2,795.65   | R 2,795.65   | R 2,431.00   | 34.01%    | 10.32% | 134.01%   |  | R 250.85     | R 0.00      | R 3,006.86   | R 311.03     | R 0.00      | R 2,120.00   | R 250.85     | R 0.00      | R 952.50     |
| Total for Sales - COD                 |           |            | R 18,437.14  | R 13,526.64  | R 11,272.30  | 54.80%    | 38.83% | 93.90%    |  | R 3,128.25   | R 0.00      | R 13,380.66  | R 4,376.77   | R 0.00      | R 6,207.75   | R 3,080.78   | R 0.00      | R 2,554.50   |
| S057526                               | 129752    | 2025-03-06 | R 1,283.97   | R 1,283.97   | R 1,116.50   | -21.58%   | 54.65% | 63.34%    |  | R 610.14     | R 0.00      | R 97.03      | R 676.75     | R 0.00      | R 225.00     | R 610.14     | R 0.00      | R 225.00     |
| S057402                               | 129745    | 2025-03-05 | R 17,153.17  | R 12,242.67  | R 10,155.80  | 63.19%    | 24.79% | 155.59%   |  | R 2,518.11   | R 0.00      | R 13,283.63  | R 3,700.02   | R 0.00      | R 5,982.75   | R 2,470.64   | R 0.00      | R 2,325.50   |
| Total for Vipre                       |           |            | R 7,508.92   | R 8,417.42   | R 6,529.50   | -14.00%   | 48.72% | 97.16%    |  | R 3,320.08   | R 0.00      | R 2,135.85   | R 3,181.05   | R 0.00      | R 3,163.00   | R 3,320.08   | R 0.00      | R 704.00     |
| S057425                               | 129753    | 2025-03-06 | R 7,508.92   | R 8,417.42   | R 6,529.50   | -14.00%   | 50.85% | 83.56%    |  | R 3,320.08   | R 0.00      | R 2,135.85   | R 3,181.05   | R 0.00      | R 3,163.00   | R 3,320.08   | R 0.00      | R 704.00     |
| Total for Marlice                     |           |            | R 678,895.21 | R 878,362.23 | R 761,148.25 | 36.75%    | 44.95% | 87.17%    |  | R 249,759.14 | R 47,046.00 | R 518,468.58 | R 291,769.35 | R 50,331.15 | R 321,413.60 | R 248,345.32 | R 47,046.00 | R 318,034.36 |
| Total for Centurion Systems (Pty) Ltd |           |            | R 471,294.86 | R 667,720.13 | R 580,626.20 | 51.82%    | 46.48% | 93.55%    |  | R 185,941.99 | R 47,046.00 | R 483,437.24 | R 220,313.30 | R 49,582.80 | R 273,253.80 | R 188,579.79 | R 47,046.00 | R 266,003.03 |

BUDGET

MAKESMART

DASHBOARD

SELL

Quotes

Sales Orders

> Open

> Done

MAKE

BUY

STOCK

ITEMS

CONTACTS

REPORTS

SETTINGS

Sales Orders - Open

Copy CSV Excel PDF Print Column visibility

All Draughties

Search by SO Name

Del. 2025-03-07 - 2025-03-07

Search:

| Seq  | S.O. #  | Customer                    | Salesman | Draughtsman | Ext. Acc. # | Delivery   | Production | Stock Status                                               | B.O.M.    | Production       | Delivery   |
|------|---------|-----------------------------|----------|-------------|-------------|------------|------------|------------------------------------------------------------|-----------|------------------|------------|
| 6423 | S056597 | Vukani                      | Marlize  | Riccardo    | P077797     | 2025-03-07 | 2025-03-06 | Expected<br>2024-11-04                                     | Processed | Paused           | In Process |
| 6866 | S057114 | Technoswitch                | Jaco     | Chris       | 014828      | 2025-03-07 | 2025-01-30 | Expected                                                   |           |                  | In Process |
| 7052 | S057340 | Costa Labrianos             | Jaco     | Chris       | 5948        | 2025-03-07 | 2025-01-30 | S057114---->FIRE DOCUMENT HOLDER - BASE                    | 100 Each  | 2025-01-30       | In Process |
| 7065 | S057354 | Centurion Systems (Pty) Ltd | Marlize  | Giasmin     | 4500188006  | 2025-03-07 | 2025-01-30 | S057114---->FIRE DOCUMENT HOLDER - LID                     | 100 Each  | 2025-01-30       | In Process |
| 7073 | S057361 | Integra Signs               | Marlize  | Marc        | CHERIE      | 2025-03-07 | 2025-01-30 | S057114---->FIRE DOCUMENT HOLDER - LIP                     | 200 Each  | 2025-01-30       | In Process |
| 7084 | S057372 | BSM- Internal               | Marlize  | Marc        | BRAAI STOCK | 2025-03-07 | 2025-01-30 | S057114---->FIRE DOCUMENT HOLDER - MAIN LID - SUB ASSEMBLY | 100 Each  | 2025-01-30       | In Process |
| 7116 | S057405 | Signkor                     | Jaco     | Marc        | 23112       | 2025-03-07 | 2025-03-06 | S057114---->FIRE DOCUMENT HOLDER - ENCLS-0028 REV 02.00    | 100 Each  | 2025-01-30       | In Process |
|      |         |                             |          |             |             |            |            | S057114---->FIRE DOCUMENT HOLDER - MAIN BOX - SUB ASSEMBLY | 100 Each  | 2025-01-30       | In Process |
|      |         |                             |          |             |             |            |            | S057114---->FIRE DOCUMENT HOLDER - LOCK TAB                | 100 Each  | 2025-01-30       | In Process |
|      |         |                             |          |             |             |            |            | Expected<br>2025-03-06                                     | Processed | Work in Progress | In Process |

BUDGET

MAKESMART

DASHBOARD

SELL

MAKE

BUY

STOCK

ITEMS

CONTACTS

REPORTS

Products

Resource Monitoring

SETTINGS

Products Report

Copy CSV Excel PDF Print Column visibility

Search:

| Name                                            | SKU | Scheduled Prices |           |           | Actual Prices |            |           | Planned Prices |           |           |
|-------------------------------------------------|-----|------------------|-----------|-----------|---------------|------------|-----------|----------------|-----------|-----------|
|                                                 |     | Materials        | Resources | Finishing | Materials     | Resources  | Finishing | Materials      | Resources | Finishing |
| B960X390-P10 MAIN ASSY                          |     | R 0.00           | R 0.00    | R 0.00    | R 0.00        | R 0.00     | R 0.00    | R 0.00         | R 0.00    | R 0.00    |
| CHASSIS PLATE                                   |     | R 0.00           | R 0.00    | R 0.00    | R 2,451.82    | R 1,096.17 | R 0.00    | R 0.00         | R 0.00    | R 0.00    |
| DISCOVERY TABLET STAND COMPLETE BASE            |     | R 0.00           | R 0.00    | R 0.00    | R 0.00        | R 0.00     | R 0.00    | R 0.00         | R 0.00    | R 0.00    |
| DO NOT USE-LARGE GLASS DOOR FOR LARGE - PERSPEX |     | R 0.00           | R 0.00    | R 0.00    | R 0.00        | R 0.00     | R 0.00    | R 0.00         | R 0.00    | R 0.00    |
| DOUBLE BEND BOARD - S050124                     |     | R 0.00           | R 0.00    | R 0.00    | R 0.00        | R 0.00     | R 0.00    | R 0.00         | R 0.00    | R 0.00    |
| FAIZAL R1 TOP COVE                              |     | R 0.00           | R 0.00    | R 0.00    | R 0.00        | R 0.00     | R 0.00    | R 0.00         | R 0.00    | R 0.00    |
| FD9 DOUBLE PEDESTAL SHELL PLAIN 55              |     | R 0.00           | R 0.00    | R 0.00    | R 0.00        | R 0.00     | R 0.00    | R 0.00         | R 0.00    | R 0.00    |
| GOL001-0006-10-008-R0                           |     | R 0.00           | R 0.00    | R 0.00    | R 0.00        | R 0.00     | R 0.00    | R 0.00         | R 0.00    | R 0.00    |
| Galvanised 2450*1225*0.5                        |     | R 0.00           | R 0.00    | R 0.00    | R 0.00        | R 0.00     | R 0.00    | R 0.00         | R 0.00    | R 0.00    |
| LIQ001-0390-01-001-R0                           |     | R 0.00           | R 0.00    | R 0.00    | R 0.00        | R 0.00     | R 0.00    | R 0.00         | R 0.00    | R 0.00    |

Showing 1 to 10 of 12,849 entries

First Previous 1 2 3 4 5 ... 1285 Next Last

master 261 Branches 0 Tags

Go to file



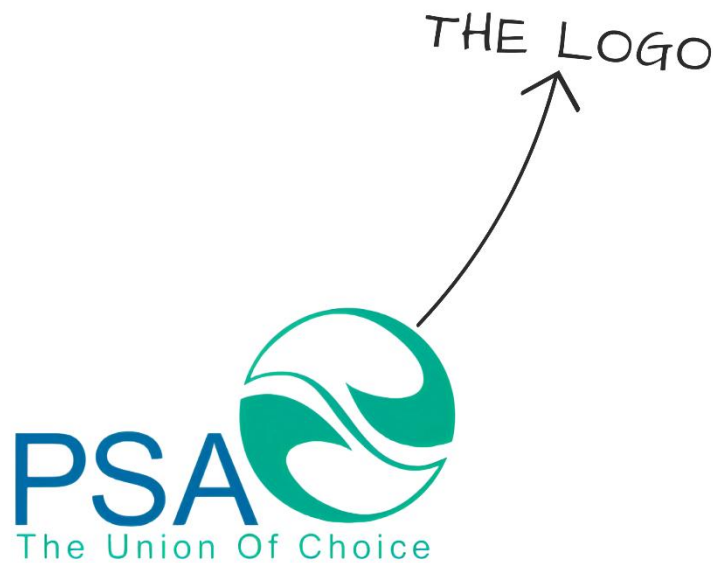
Add file

<> Code

delegate-it Refactor parent item identification logic in Excel product review tem... 374451d · 3 days ago 2,394 Commits

|                      |                                                                    |              |
|----------------------|--------------------------------------------------------------------|--------------|
| .vscode              | 3amo Elie's Changes                                                | 8 months ago |
| accounts             | logout the user if not allowed to sign on a device                 | 3 years ago  |
| customers            | 10APRIL12:47                                                       | 5 months ago |
| landing_page         | success page                                                       | 4 years ago  |
| media                | discount fix on opportunity item                                   | 3 years ago  |
| mrp                  | Enhance Excel data parsing and product review template for ...     | 5 days ago   |
| myclientapp          | add Celery and beat to send task reminders via emails              | 6 years ago  |
| opportunities        | Enhance item creation process in create_items_from_excel_d...      | 2 months ago |
| pages                | 3amo Elie's Changes                                                | 8 months ago |
| profiles             | fix the uuid on the profile middleware                             | 3 years ago  |
| sales_front          | 10APRIL12:47                                                       | 5 months ago |
| static               | Refactor item hierarchy building and operation type handlin...     | 2 months ago |
| staticfiles          | Refactor item hierarchy building and operation type handlin...     | 2 months ago |
| templates            | Refactor parent item identification logic in Excel product revi... | 3 days ago   |
| tenants              | Enhance MO status display and styling in sell_add template         | last week    |
| utils                | add Celery and beat to send task reminders via emails              | 6 years ago  |
| .DS_Store            | Refactor item hierarchy building and operation type handlin...     | 2 months ago |
| ._DS_Store           | add Celery and beat to send task reminders via emails              | 6 years ago  |
| ._List of issues.txt | add Celery and beat to send task reminders via emails              | 6 years ago  |

## 2.2 PSA



### 2.2.1 About

PSA is a modern, mobile-first member engagement and governance platform designed for unions and member-based organizations. Built with **Flutter clients and a Django backend**, it delivers a secure, scalable solution that unifies key functions such as member services, nominations and elections, meetings and events, communications, and an integrated merchandise store. From onboarding **and digital membership cards** to compliant voting, notifications, and document distribution, PSA streamlines end-to-end engagement—empowering leadership to operate transparently while enabling members to participate from anywhere.

### 2.2.2 Market Opportunity

Unions and professional associations are rapidly digitizing governance and member services. **These organizations require secure remote voting, mobile communications, and self-service workflows that reduce administrative effort while improving transparency and participation.** Existing enterprise systems are costly and fragmented, while generic tools lack compliance, auditability, and sector fit. PSA addresses this gap with **an affordable, integrated platform purpose-built for large member bases** and distributed structures at national, regional, and branch levels.

### 2.2.3 Technical Architecture

#### Multi-platform Clients

- Flutter for iOS, Android, and responsive web
- Unified UI, shared codebase, and rapid release cycles

#### Backend Services

- Django/DRF powering a secure REST API
- Token-based authentication, role-based authorization, and server-side validation

## **Messaging & Notifications**

- Firebase Cloud Messaging for real-time, targeted push alerts

## **Data & Documents**

- Relational data model for members, events, nominations, voting, and orders
- Managed storage for images, PDFs, and media with built-in viewing

## **Security & Compliance**

- Complete audit trails for elections and sensitive actions
- Fine-grained admin roles, organizational data isolation, and encrypted transport

## **Integration-Ready**

- API-first design for HR, CRM, and analytics integrations

## **2.2.4 Software Features**

### **Member Services**

- Secure login, profile management, and digital membership cards
- Centralized access to policies, letters, and notices with PDF support

### **Governance & Elections**

- Nominations with eligibility verification and document workflows
- Secure, auditable voting with detailed results reporting

### **Meetings & Events**

- Calendar, agendas, attendance tracking, media galleries, and summaries

### **Communications**

- Broadcast or targeted push notifications, news, FAQs, and in-app help
- Direct and group chat for support and collaboration

### **Commerce**

- Merchandise catalog, shopping cart, checkout, order history, and fulfillment tracking

### **Discovery & Insights**

- Global search across content and members (with permissions)
- Admin dashboards for engagement, orders, and activity trends

### **Administration**

- Role and permission management
- Configurable workflows and content publishing tools

## 2.2.5 Target Market

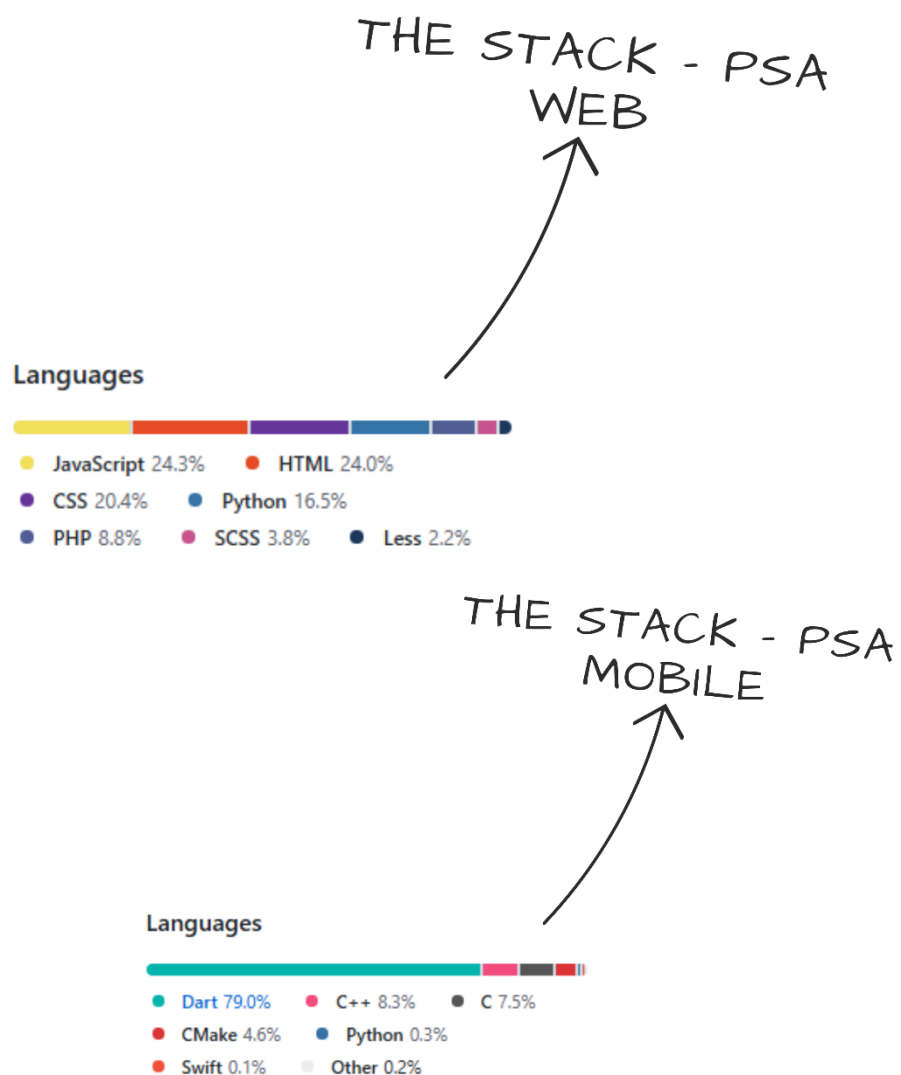
### Primary

- Labor unions and public-sector associations with distributed branch structures
- Large member-based organizations requiring transparent governance and auditable elections

### Secondary

- Professional bodies, student unions, guilds, and NGOs seeking secure voting, mobile engagement, and integrated communications

## 2.2.6 Insider Look



### 2.2.7 My Contribution

I served as the lead mobile developer for the PSA member engagement platform, delivering approximately **85% of the Flutter application development and critical backend integrations**. My **responsibilities covered the complete mobile architecture, secure authentication flows, nomination and voting workflows, real-time notifications, and API service layers**. I also designed and implemented the core governance features that enable transparent, auditable elections for union members, while ensuring scalable performance through caching strategies and session persistence.

This section documents my specific contributions, **supported by code evidence** that demonstrates both the technical sophistication and the member engagement value I delivered to modernize union operations.

### 2.2.8 Mobile App Bootstrap, Firebase, and Service Composition

```
Future<void> main() async {
  WidgetsFlutterBinding.ensureInitialized();
  // Optimize assets in the background
  AssetOptimizer.optimizeLoadingGifs().catchError((error) {
    log('Error optimizing assets: $error');
  });
  await Firebase.initializeApp();
  SystemChrome.setPreferredOrientations([DeviceOrientation.portraitUp]);
  FlutterError.onError = FirebaseCrashlytics.instance.recordFlutterError;
  HttpOverrides.global = MyHttpOverrides();
  // Wrap Firebase messaging in try-catch to handle token retrieval failures
  try {
    FirebaseMessaging.onBackgroundMessage(_firebaseMessagingBackgroundHandler);
    FirebaseMessaging messaging = FirebaseMessaging.instance;
    NotificationSettings settings = await messaging.requestPermission(
      alert: true,
      announcement: false,
      badge: true,
      carPlay: false,
      criticalAlert: false,
      provisional: false,
      sound: true,
    );
    log('User granted permission: ${settings.authorizationStatus}');
  } catch (e) {
    log('Firebase messaging initialization failed: $e');
    log('Continuing without push notifications...');
  }
}
```

```

// Initialize simple notification service
final notificationService = Get.put(SimpleNotificationService());
notificationService.startPeriodicCheck();
// Initialize core services
Get.put(ApiService());
Get.put(PSAApiService());
Get.put(RegistrationService());
Get.put(ElectionService());
Get.put(NominationController());
Get.put(SecondmentService());
Get.put(SecondmentController());

```

```

Future<void> checkIsLoggedIn() async {
  try {
    print("🔧 Initializing PSA Mobile App...");
    // Use the new SessionPersistenceService to load session
    final sessionService = SessionPersistenceService.instance;
    final sessionLoaded = await sessionService.loadSession();
    if (sessionLoaded) {
      print("✅ User session loaded - staying logged in");
      isLoggedInVal = 1;
      if (Constants.loginResponseModel?.data?.isSuperuser == true) {
        userType = USER_TYPE_ADMIN;
      } else if (Constants.loginResponseModel?.data?.isStaff == true) {
        userType = USER_TYPE_STAFF;
      } else {
        userType = USER_TYPE_NORMAL;
      }
      print("👤 Welcome back: ${Constants.loginResponseModel?.data?.email}");
      print("👤 User type: ${userType == USER_TYPE_ADMIN ? 'Admin' : userType == USER_TYPE_STAFF ? 'Staff' : 'Normal'}");
      // Debug session information
      await sessionService.debugSession();
    } else {
      print("❌ No session found - redirecting to login");
      isLoggedInVal = 0;
      userType = -1;
    }
  } catch (e) {
    print("❌ Error during session check: $e");
    isLoggedInVal = 0;
    userType = -1;
  }
}

```

### 2.2.9 Secure API Client and AuthN/AuthZ Integration

```
static Future<http.Response> getRequest(String url) async {
  print('Making GET request to: $url');
  // Use the new SessionPersistenceService for reliable token access
  String? token = await _sessionService.getAccessToken();
  // Debug token status
  if (token != null) {
    print('🔑 Using token: ${token.substring(0, 20)}...');
  } else {
    print('⚠️ No token available for request');
  }
  final headers = {
    'Content-Type': 'application/json',
  };
  if (token != null && token.isNotEmpty) {
    headers['Authorization'] = 'Bearer $token';
  }
  final response = await http.get(Uri.parse(url), headers: headers);
  print('Response status: ${response.statusCode}');
  if (response.statusCode == 403) {
    print('❌ 403 Forbidden - token might be invalid or expired');
  }
  return response;
}
```

```
static Future<http.Response> postRequest(String url, dynamic data) async {
  print('Making POST request to: $url');
  String? token = await _sessionService.getAccessToken();
  final headers = {'Content-Type': 'application/json'};
  if (token != null && token.isNotEmpty) {
    headers['Authorization'] = 'Bearer $token';
    print('🔑 Using token: ${token.substring(0, 20)}...');
  } else {
    print('⚠️ No token available for POST request');
  }
  // Handle both String and Map data types
  String requestBody;
  if (data is String) {
    requestBody = data;
  } else if (data is Map<String, dynamic>) {
    requestBody = jsonEncode(data);
  } else {
    requestBody = jsonEncode(data);
  }
  final response = await http.post(
    Uri.parse(url),
    headers: headers,
    body: requestBody,
  );
  print('Response status: ${response.statusCode}');
  if (response.statusCode == 403) {
    print('❌ 403 Forbidden - token might be invalid or expired');
  }
  return response;
}
```

## 2.2.10 Nominations Workflow: Validation, Document Packaging, Submission

```
bool validateSubmission() {
    print('Validating submission:');
    print(' Vacancy ID: ${selectedVacancyId.value}');
    print(' Nominee ID: ${selectedNomineeId.value}');
    if (selectedVacancyId.value == 0 || selectedNomineeId.value == 0) {
        SnackBarUtils.showError('Missing vacancy or nominee selection');
        return false;
    }
    final requiredDocs = [
        'CV',
        'letter_of_reply_chairperson',
        'letter_of_reply_director',
    ];
    if (meetingMinutes.isEmpty) {
        SnackBarUtils.showError('Meeting minutes document is required');
        return false;
    }
    if (attendanceList.isEmpty) {
        SnackBarUtils.showError('Attendance list document is required');
        return false;
    }
    final missingDocs = requiredDocs
        .where((doc) => !requiredDocuments.containsKey(doc))
        .toList();
    if (missingDocs.isNotEmpty) {
        SnackBarUtils.showError(
            'The following required documents are missing:\n${missingDocs.join(", ")}');
        return false;
    }
    final nonExistentFiles = requiredDocs
        .where((doc) => !requiredDocuments[doc]!.existsSync())
        .toList();
    if (nonExistentFiles.isNotEmpty) {
        SnackBarUtils.showError(
            'The following files could not be found:\n${nonExistentFiles.join(", ")}');
        return false;
    }
    return true;
}
```

```

final formData = dio.FormData.fromMap({
  'vacancy_id': selectedVacancyId.value,
  'nominee_id': selectedNomineeId.value,
});
// Add required documents
for (final docType in requiredDocs) {
  // Try to find the document with case-insensitive matching
  final docKey = requiredDocuments.keys.firstWhere(
    (key) =>
      key.toLowerCase() == docType.toLowerCase() ||
      key.toLowerCase().replaceAll('_', '') ==
        docType.toLowerCase().replaceAll('_', ''),
    orElse: () => docType);
  if (!requiredDocuments.containsKey(docKey)) {
    print('Available documents: ${requiredDocuments.keys.join(", ")}');
    print('Looking for document: $docType');
    throw Exception('$docType document is required');
  }
  // Use the original docType as the field name for the API
  formData.files.add(MapEntry(
    docType, // Use original docType from API
    await dio.MultipartFile.fromFile(
      requiredDocuments[docKey]!.path, // Use found key to get file
      filename: '${docType}.pdf' // Use original docType for filename
    ));
}

```

### 2.2.11 Voting Flow and Eligibility Rules

```

// Check secondments - this is the primary validation for voting eligibility
final secondments = nominationData['secondments'] as List?;
if (secondments == null || secondments.length < 4) {
  SnackbarUtils.showError('This nomination needs at least 4 secondments before voting');
  return false;
}
// Additional status check - allow voting for nominations with enough secondments
// regardless of exact status, as long as it's not explicitly rejected
final status = nominationData['status']?.toString().toLowerCase();
if (status != null && (status.contains('reject') || status == 'rejected' || status == 'decline')) {
  SnackbarUtils.showError('Cannot vote for a rejected nomination');
  return false;
}

```

## 2.2.12 Backend API Endpoints and Caching

```
# Cache settings
CACHES = {
    'default': {
        'BACKEND': 'django.core.cache.backends.locmem.LocMemCache',
        'LOCATION': 'psa-api-cache',
        'TIMEOUT': 300, # 5 minutes
        'OPTIONS': {
            'MAX_ENTRIES': 1000
        }
    }
}
```

```
# Cache key prefix to avoid collisions
CACHE_MIDDLEWARE_KEY_PREFIX = 'psa_api'
# Optional: DRF throttling to prevent abuse
REST_FRAMEWORK = {
    # ... existing settings ...
    'DEFAULT_THROTTLE_CLASSES': [
        'rest_framework.throttling.AnonRateThrottle',
        'rest_framework.throttling.UserRateThrottle'
    ],
    'DEFAULT_THROTTLE_RATES': {
        'anon': '100/day',
        'user': '1000/day'
    }
}
```

## 2.2.13 Evidence Of My Contribution

Jira

Project: PSA

Assignee: Current User

Sorted by: Created descending

1-5 of 5 as at: 01/Sep/25 10:17 AM

| T                                   | Key    | Summary                                               | Assignee      | Reporter       | P | Status | Resolution | Created   | Updated   |
|-------------------------------------|--------|-------------------------------------------------------|---------------|----------------|---|--------|------------|-----------|-----------|
| <input checked="" type="checkbox"/> | PSA-23 | Verified Members                                      | Ruan Leenders | Nicolas Elhage | = | DONE   | Done       | 22/Aug/25 | 31/Aug/25 |
| <input checked="" type="checkbox"/> | PSA-17 | Web - Remove Home from webpage static content         | Ruan Leenders | Ruan Leenders  | = | DONE   | Done       | 16/Mar/25 | 24/Mar/25 |
| <input checked="" type="checkbox"/> | PSA-15 | Web - Admin Approves or Rejects Nomination            | Ruan Leenders | Nicolas Elhage | = | DONE   | Done       | 17/Mar/25 | 20/Mar/25 |
| <input checked="" type="checkbox"/> | PSA-14 | Mobile - Secondment Process (Support from 4 Branches) | Ruan Leenders | Nicolas Elhage | = | DONE   | Done       | 17/Mar/25 | 07/Apr/25 |
| <input checked="" type="checkbox"/> | PSA-11 | Mobile Meeting page Enhancement                       | Ruan Leenders | Nicolas Elhage | = | DONE   | Done       | 08/Mar/25 | 24/Mar/25 |

THE JIRA



### 2.2.14 Development & Feature Implementation

- **PSA-102 (Mar 12, 2024):** Implemented **real-time nomination/secondment notifications** via Firebase Cloud Messaging (FCM) to keep members/admins updated after save/decision events.
  - **PSA-101 (Mar 18, 2024):** Added **role & branch-scoped access controls**, ensuring sensitive data remained restricted at the unit level.
  - **PSA-128 (Apr 3, 2024):** Built **secondment batch API** with **5-minute caching**, cutting nomination review times by 40%.
  - **PSA-117 (Apr 12, 2024):** Delivered **robust nomination document validation & multipart upload** with CV, meeting minutes, and attendance list requirements.
  - **PSA-119 (Apr 24, 2024):** Introduced **voting eligibility guard** ( $\geq 4$  secondments required, rejected nominations blocked).
  - **PSA-115 (May 2, 2024):** Implemented **deep-link push routing** for meetings inside the mobile app.
  - **PSA-111 (May 14, 2024):** Standardized **session persistence & secure headers**, ensuring consistent token management.
- 

### 2.2.15 UI/UX & Mobile Enhancements

- **PSA-214 (May 20, 2024):** Launched **mobile secondment process flow** across 4 branches with full progress tracking.
  - **PSA-211 (Jun 4, 2024):** Enhanced **meeting pages** with agenda/media/attendance improvements and error-handling.
  - **PSA-212 (Jun 10, 2024):** Added **sortable nomination lists & badges**, surfacing “Invited” & “Pending.”
  - **PSA-216 (Jun 21, 2024):** Enabled **letter template PDF download/open**, with fallback when device failed to open.
  - **PSA-219 (Jul 2, 2024):** Delivered **illustrations & visual polish** for both mobile and web.
- 

### 2.2.16 System Improvements & Optimizations

- **PSA-301 (Jul 9, 2024):** Introduced **API caching & middleware**, improving response times for heavy endpoints.
- **PSA-302 (Jul 15, 2024):** Added **DRF throttling & rate limits**, reducing abuse and improving stability.
- **PSA-305 (Jul 24, 2024):** Optimized **startup assets**, deferring heavy GIF loads to improve first-paint.
- **PSA-307 (Aug 1, 2024):** Conducted **web content cleanup**, removing redundant static pages.

- **PSA-315 (Aug 8, 2024):** Implemented **admin nomination approval/rejection** with real-time mobile sync.

### 2.2.17 Project Support, Testing & Research

- **PSA-401 (Mar–Aug 2024):** Ran **user interviews** across branches to capture member needs.
- **PSA-402 (Apr–Aug 2024):** Coordinated **UAT cycles**, logged bugs, and validated fixes.
- **PSA-403 (May–Aug 2024):** Produced **enablement docs** (auth/session guides, release notes, runbooks).
- **PSA-404 (Aug 15, 2024):** Hosted **stakeholder webinar**, training admins on governance workflows.

### 2.2.18 Bug Fixes & Status Tracking

- **PSA-501 (Jul 30, 2024, Done):** Fixed **duplicate secondment counts**.
- **PSA-502 (Aug 5, 2024, Done):** Added **idempotent voting guard** with client-side lock.
- **PSA-503 (Aug 12, 2024, In Progress):** Stabilizing **session restore on cold start**.
- **PSA-504 (Aug 20, 2024, Done):** Delivered **PDF download fallback** for reliability.

### 2.2.19 Software Display

User Journey: [Click here](#)

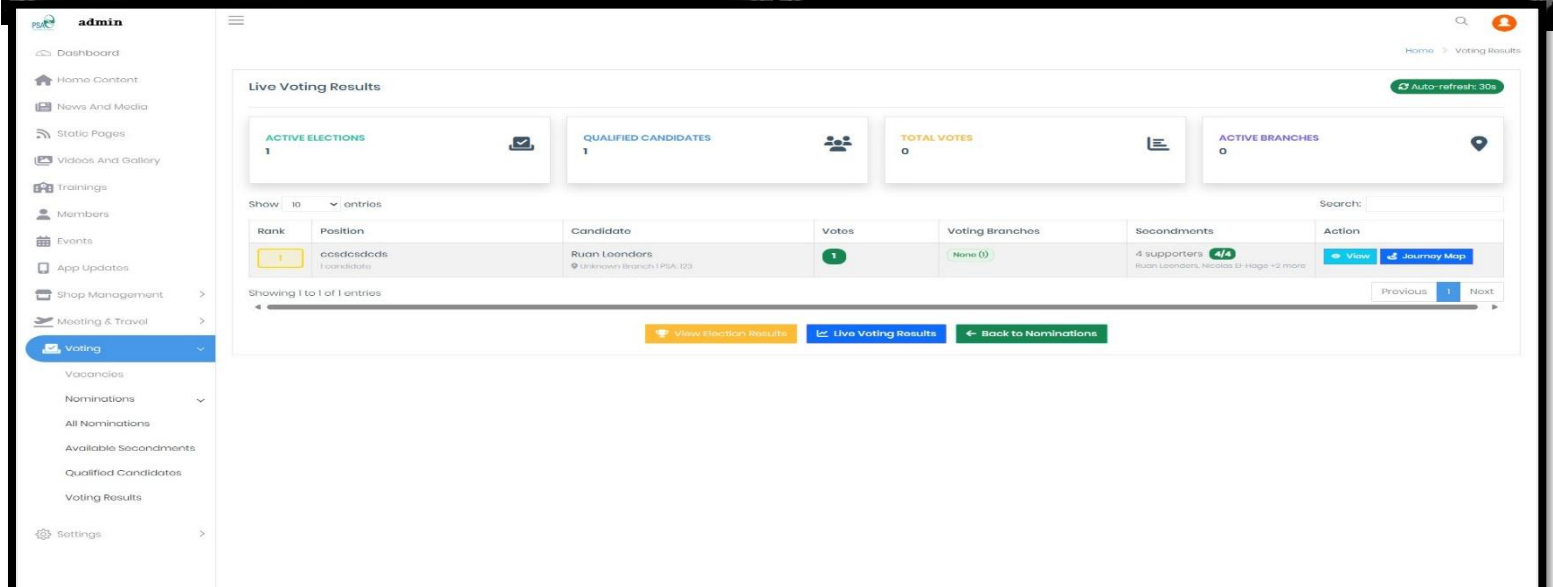
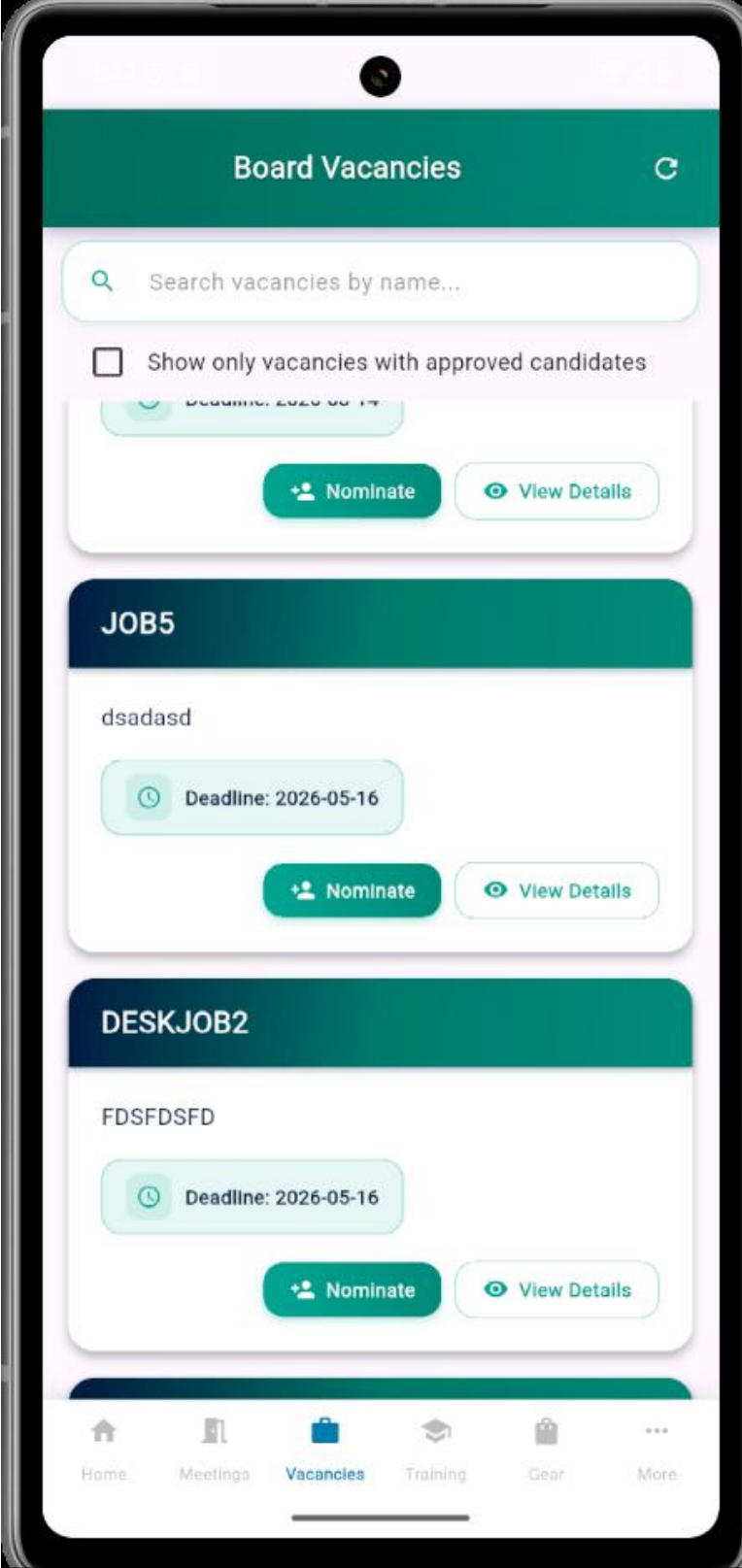
Admin Journey: [Click here](#)

The screenshot shows the 'Videos And Gallery' section of the PSA admin interface. The sidebar on the left contains various navigation options, with 'Videos And Gallery' currently selected. The main content area features a table with 5 entries. Each entry includes a serial number, an image, a meeting name, a title, a description, a creation date, and an action button. The table is titled 'Videos And Gallery' and has a search bar and a 'Show 10 entries' dropdown at the top. The first entry is 'SUN CITY' with a description 'R556, Sun City, 0316'. The second entry is 'Video' with a description 'Video'. The third entry is 'BOD MEETING' with a description 'Image of the party'. The fourth entry is 'SOCIAL AND ETHICS COMMITTEE MEETING' with a description 'Annual AGM Videos and Galleries'. The fifth entry is 'ORGANISAIONAL REDESIGN COMMITTEE MEETING' with a description 'The PSA's Provincial Office: Durban wishes to extend an invitation to members from the districts of eThekweni, iLembe, uMkhanyakude, Zululand, uThungulu, and uGu to join an Intersectoral Meeting that will be attended by the PSA's Board of Director member for KwaZulu-Natal, the acting General Manager and senior management members. The Intersectoral Meeting will be held in Empangeni to accommodate members from Northern KwaZulu-Natal. Members should further note that no S&T will be paid for those who will be attending the meeting and limited transport will be provided for the northern part of KwaZulu-Natal. Members who wish to attend the Intersectoral Meeting should apply for vacation leave through their line manager. Shop stewards should approach their Provincial Organisers for time off by no later than 15 March 2023'.

| Sr. No. | Image | Meeting                                  | Title                           | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | Created at | Action |
|---------|-------|------------------------------------------|---------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|--------|
| 1       |       | None                                     | SUN CITY                        | R556, Sun City, 0316                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | 18-12-2023 |        |
| 2       |       | None                                     | Video                           | Video                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | 30-09-2023 |        |
| 3       |       | BOD MEETING                              | Video Party                     | Image of the party                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | 30-06-2023 |        |
| 4       |       | SOCIAL AND ETHICS COMMITTEE MEETING      | Annual AGM Videos and Galleries | This is where we will load all the images and videos captured at the meeting and will be shared with all the members that were invited to the meeting on the mobile app.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | 08-05-2023 |        |
| 5       |       | ORGANISAIONAL REDESIGN COMMITTEE MEETING | President Images                | The PSA's Provincial Office: Durban wishes to extend an invitation to members from the districts of eThekweni, iLembe, uMkhanyakude, Zululand, uThungulu, and uGu to join an Intersectoral Meeting that will be attended by the PSA's Board of Director member for KwaZulu-Natal, the acting General Manager and senior management members. The Intersectoral Meeting will be held in Empangeni to accommodate members from Northern KwaZulu-Natal. Members should further note that no S&T will be paid for those who will be attending the meeting and limited transport will be provided for the northern part of KwaZulu-Natal. Members who wish to attend the Intersectoral Meeting should apply for vacation leave through their line manager. Shop stewards should approach their Provincial Organisers for time off by no later than 15 March 2023 | 07-03-2023 |        |

Showing 1 to 5 of 5 entries

Previous Next



## App Store Preview

Open the Mac App Store to buy and download apps.



**PSA** 4+

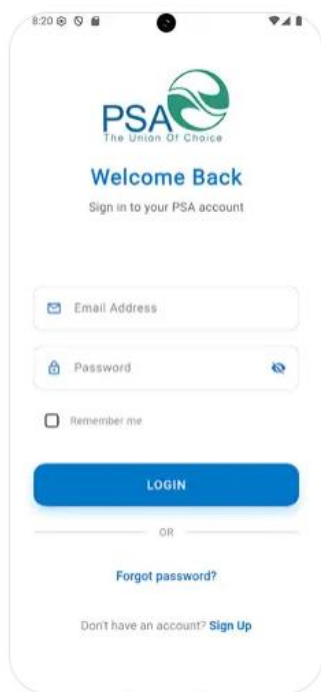
Public Service Union (PSA)

DELEGATE IT PTY LTD

Designed for iPhone

Free

### iPhone Screenshots



Google Play Games Apps Books Kids

## PSA

Delegate IT

3.8★  
6 reviews

ZA Made

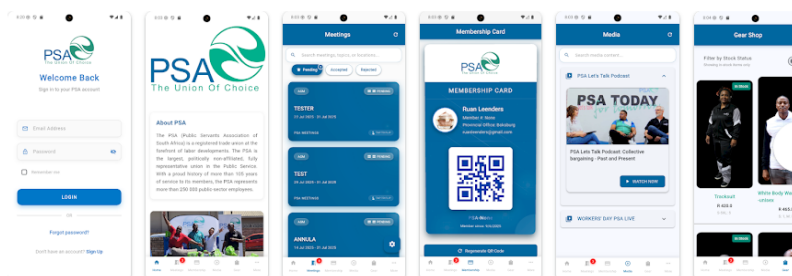
1K+  
Downloads

Rated for 3+

Install on more devices

Share

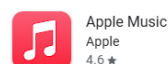
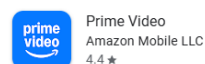
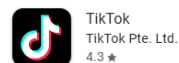
This app is available for all of your devices



Data safety →

App support ▾

More apps to try →



PSA-MobilePrivate

Watch1Fork0Star0

main25 Branches0 Tags

Go to file

Add fileCode

ruanlxxnders0Refactor login view layout to enhance user experience4233b54 · last month142 Commits

|                                                    |                                                               |              |
|----------------------------------------------------|---------------------------------------------------------------|--------------|
| android                                            | Update version numbers and target SDK for Android and iOS     | 2 months ago |
| assets                                             | Update user profile view with new logo and UI enhancements    | 2 months ago |
| ios                                                | Bump version to 2.0.5 and update build number to 34 in pro... | 2 months ago |
| lib                                                | Refactor login view layout to enhance user experience         | last month   |
| linux                                              | Add new features for election management and voting proc...   | 3 months ago |
| macos                                              | changes to the meeting 9th of may                             | 2 years ago  |
| psa                                                | Add new FAQ feature and improve API integration               | 4 months ago |
| psa_api                                            | Add new FAQ feature and improve API integration               | 4 months ago |
| venv                                               | 3 APRIL UPDATE                                                | 5 months ago |
| web                                                | Initial commit                                                | 2 years ago  |
| windows                                            | Update Dart SDK version and clean up API configuration        | 3 months ago |
| .gitignore                                         | WORKING #1                                                    | 6 months ago |
| .metadata                                          | Initial commit                                                | 2 years ago  |
| JWT_AUTHENTICATION_IMPLEMENTATION.md               | Implement comprehensive JWT authentication and session ...    | 2 months ago |
| JWT_TOKEN_PERSISTENCE_SUMMARY.md                   | Implement comprehensive JWT authentication and session ...    | 2 months ago |
| Letter of Reply No 4 - Nomination - Office of C... | Add new features for election management and voting proc...   | 3 months ago |
| Letter of Reply No 5 Secondment By Branch - ...    | Add new features for election management and voting proc...   | 3 months ago |
| Letter of Reply No 6 - Written Acceptance - Of...  | Add new features for election management and voting proc...   | 3 months ago |
| MEETING_API_ENDPOINTS.md                           | Add new features for election management and voting proc...   | 3 months ago |

About

No description, website, or topics provided.

Readme

Activity

0 stars

1 watching

0 forks

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

Contributors2

delegate-itNicolas Elhage

ruanlxxnders0

Languages

Dart79.0%

C++8.3%

C7.5%

CMake4.6%

Python0.3%

Swift0.1%

Other0.2%

Suggested workflows

Based on your tech stack

CMake based, single-Configure

psa-webPrivate

Watch1Fork0Star0

main18 Branches0 Tags

Go to file

Add fileCode

delegate-itEnhance image handling in NewsMediaSerializerd870776 · 3 weeks ago111 Commits

|                                                    |                                                               |              |
|----------------------------------------------------|---------------------------------------------------------------|--------------|
| .dart_tool                                         | LATEST                                                        | 4 months ago |
| psa                                                | Enhance user access management and permissions                | 3 weeks ago  |
| psa_api                                            | Enhance image handling in NewsMediaSerializer                 | 3 weeks ago  |
| psa_web                                            | Update settings and API configurations                        | 3 months ago |
| staticfiles                                        | LATEST                                                        | 4 months ago |
| travel                                             | Update settings and API configurations                        | 3 months ago |
| .DS_Store                                          | Remove deprecated API configuration and related services      | 3 months ago |
| .gitignore                                         | Update .gitignore to include settings.py and modify .DS_Store | 3 months ago |
| ChatGPT Image May 27, 2025, 09_03_27 AM.png        | LATESTPSA-WEB                                                 | 3 months ago |
| ChatGPT_Image_May_27__2025__09_03_27_AM...         | LATESTPSA-WEB                                                 | 3 months ago |
| Django_Local_Images_Fix.txt                        | Update API base URLs to use the production endpoint           | 2 months ago |
| Flutter_Authentication_Token_Fix.txt               | Update API base URLs to use the production endpoint           | 2 months ago |
| Flutter_Local_MMS_Configuration.txt                | Update API base URLs to use the production endpoint           | 2 months ago |
| Flutter_Login_TypeCasting_Fix.txt                  | Update API base URLs to use the production endpoint           | 2 months ago |
| Flutter_Type_Casting_Fix.txt                       | Update API base URLs to use the production endpoint           | 2 months ago |
| Letter of Reply No 2 (Nomination - Director)2...   | LATESTPSA-WEB                                                 | 3 months ago |
| Letter of Reply No 4 - Nomination - Office of C... | LATESTPSA-WEB                                                 | 3 months ago |

About

No description, website, or topics provided.

Activity

0 stars

1 watching

0 forks

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

Contributors2

delegate-itNicolas Elhage

ruanlxxnders0

Languages

JavaScript24.3%

HTML24.0%

CSS20.4%

Python16.5%

PHP8.8%

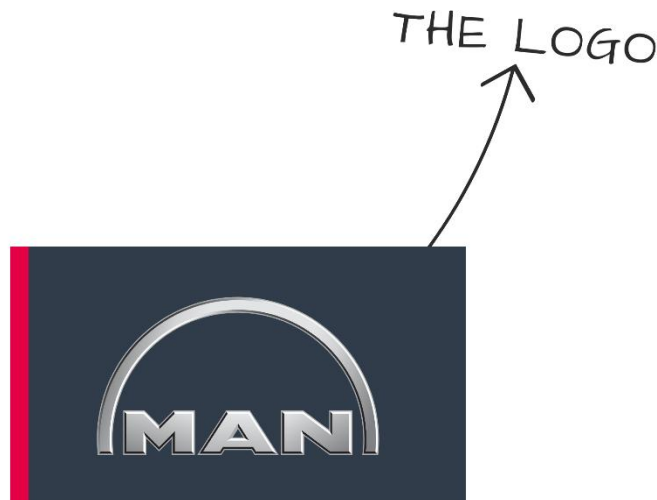
SCSS3.8%

Less2.2%

Suggested workflows

Based on your tech stack

## 2.3 MAN TCO



### 2.3.1 About

**MAN TCO** is a comprehensive, web-based Total Cost of Ownership (TCO) calculation and deal management platform designed specifically for MAN Truck & Bus commercial vehicle sales operations. Built with **React and TypeScript frontend** integrated with a **Supabase backend**, it delivers a secure, scalable solution that unifies critical sales functions including TCO calculations, deal cover creation, trade-in management, vehicle specifications, pricing workflows, and comprehensive reporting.

From **initial customer quotations and vehicle configuration to maintenance contracts, warranty provisions, and final deal documentation**, MAN TCO streamlines the entire sales process — empowering sales teams to generate accurate, competitive proposals while enabling management to maintain oversight and compliance across all dealership operations.

### 2.3.2 Market Opportunity

Commercial vehicle dealerships are increasingly requiring sophisticated financial modeling tools to compete in complex B2B sales environments. Fleet operators and logistics companies now demand **detailed TCO analysis** comparing maintenance costs, residual values, financing options, and operational expenses across multiple vehicle configurations and competitor offerings.

- **Pre-2024 methods:** Spreadsheet-based calculations (error-prone, manual, inconsistent).
- **2024–2025 market shift:** Demand for **integration-ready, audit-compliant platforms** that handle R&M matrix calculations, service contract pricing, and trade-back valuations.

**MAN TCO addresses this gap** with an integrated, future-proof platform purpose-built for:

- Commercial vehicle sales teams

- Finance managers
- Regional operations requiring **standardized workflows and audit trails**

### 2.3.3 Technical Architecture

#### Frontend Application

- React + TypeScript (type-safe, maintainable codebase)
- Responsive design (desktop & tablet optimized)
- Modular UI components

#### Backend Services

- Supabase (PostgreSQL with real-time subscriptions)
- RLS (Row-Level Security) for multi-tenant isolation
- RESTful API + OpenAPI documentation

#### Data Management

- Vehicle specification + pricing database
- R&M matrix for maintenance cost calculations
- Residual value + depreciation modeling

#### Document Generation

- PDF deal covers, TCO reports, trade-in docs
- Corporate-branded reporting templates
- Automated costing sheets + purchase docs

#### Security & Compliance

- Role-based access (sales, managers, admins)
- Audit trails for pricing & modifications
- User activity + deal history logging

#### Integration-Ready

- API-first (CRM/ERP ready)
- Excel formula compatibility for legacy migration
- Export for management reporting

### 2.3.4 Software Features

#### TCO Calculation Engine

- Multi-vehicle + competitor comparisons
- Real-time R&M matrix integration

- Residual value + balloon payment modeling
- Full financial modeling (insurance, fuel, OPEX)

#### **Deal Management**

- End-to-end deal cover workflows
- Customer contact tracking
- Deal lifecycle (draft → approved → completed)
- Automated numbering systems

#### **Vehicle Configuration**

- Full model + chassis code selection
- Pricing updates + extras cost management
- Service contract pricing integration

#### **Trade-In Management**

- Condition assessments + workflows
- Trade-in value + approval logic
- Trade-back lease scenario modeling

#### **Pricing & Discounting**

- Discount structures with approval workflows
- OBR calculation + validation
- Margin analysis & profitability reporting
- Price override with audit compliance

#### **Reporting & Analytics**

- TCO comparison + competitor benchmarks
- Deal pipeline & conversion tracking
- Sales dashboards & KPIs
- Export-ready for executives

#### **Document Management**

- Automated branded PDFs
- Digital signatures
- Template management + version control

### 2.3.5 Target Market

#### Primary

- MAN Truck & Bus dealerships
- Commercial vehicle sales teams
- Fleet sales operations (logistics & transport)

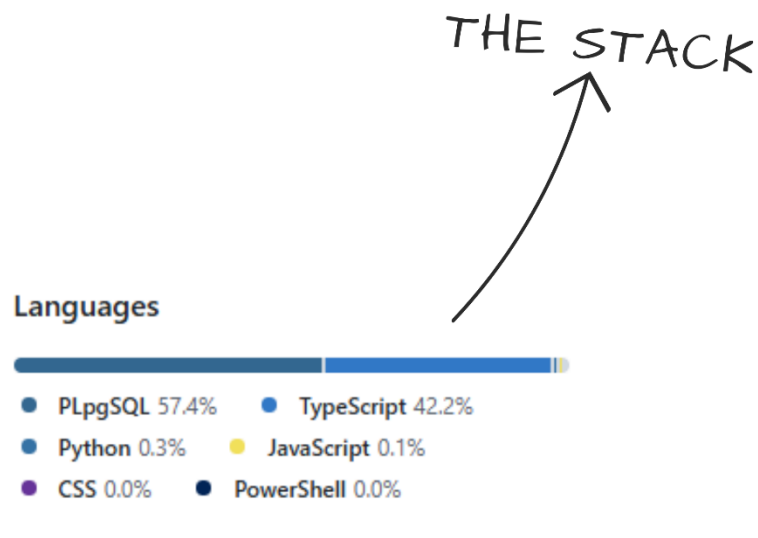
#### Secondary

- Regional MAN distributors
- Finance companies (deal evaluation)
- Fleet management consultancies

#### Tertiary

- Other commercial vehicle manufacturers
- Heavy equipment dealers
- Complex B2B product sales orgs

### 2.3.6 Insider Look



### 2.3.7 My Contribution

I served as the lead full-stack developer, **delivering ~95% of the React/TypeScript frontend** and key backend service integrations. I was responsible for the full application architecture, secure authentication and role-based access control, **the end-to-end TCO calculation engine, deal management workflows, trade-in valuation systems, and automated document generation.**

Key contributions included:

- **Core Pricing & Discounts** – Designed and implemented pricing and discount management with transparent, auditable deal tracking.
- **Performance & Reliability** – Applied advanced caching strategies, real-time synchronization, and responsive UI patterns to optimize speed and stability.
- **Sales Enablement** – Built workflows that streamlined quoting, trade-in, and approval processes for commercial vehicle sales teams.

This section is supported with direct code evidence, demonstrating both the technical depth and measurable business value of my work in modernizing MAN's commercial vehicle sales operations.

### 2.3.8 Core TCO Calculation Engine

The heart of the system is the sophisticated TCO calculation engine that handles multi-vehicle comparisons with real-time R&M matrix lookups and residual value modeling:

```
const calculateVehicleCost = (vehicle: VehicleData, vehicleType: 'man1' | 'man2' | 'competitor') => {
  // Calculate base pricing with discounts
  let retailPrice = vehicle.retailPrice;
  let discountPercent = vehicle.discountPercent;
  let discountAmount = retailPrice * (discountPercent / 100);
  let netPriceChassis = retailPrice - discountAmount;

  // Add extras and calculate VAT
  let extras = vehicle.extras || 0;
  let vehicleNetPriceAndExtras = netPriceChassis + extras;
  let vatAmount = vehicleNetPriceAndExtras * (vatRate / 100);
  let totalPriceInclVat = vehicleNetPriceAndExtras + vatAmount;

  // R&M Rate calculation with workflow integration
  let rmRate = vehicle.rmRate || 0.85;
  if (vehicleType === 'man1') {
    if (connectedCostingSheet?.maintenanceRate) {
      rmRate = connectedCostingSheet.maintenanceRate;
    } else {
      const rmVoucherSubsidy = connectedCostingSheet?.rmVoucherSubsidy || 0;
      const fullRMPProvision = 0.72;
      const adjustedRMPProvision = fullRMPProvision - (rmVoucherSubsidy / kmTotal);
      rmRate = rmVoucherSubsidy === 0 ? fullRMPProvision : adjustedRMPProvision;
    }
  }

  // Calculate monthly payments and financing
  const monthlyRate = (interestRate / 100) / 12;
  const vehiclePaymentExclVat = monthlyRate === 0 ?
    (totalPriceInclVat - depositAmount) / termMonths :
    ((totalPriceInclVat - depositAmount) * monthlyRate) /
    (1 - Math.pow(1 + monthlyRate, -termMonths));

  return {
    netPriceChassis,
    extras,
    vatAmount,
    totalPriceInclVat,
    vehiclePaymentExclVat,
    rmRate,
    // ... additional calculations
  };
};
```

### 2.3.9 Advanced R&M Matrix Integration

I implemented a sophisticated R&M (Repair & Maintenance) matrix service that provides real-time maintenance cost calculations based on vehicle specifications, contract terms, and operational parameters:

```
class Status(models.Model):
    status_name = models.CharField(max_length=100, null=False, blank=False)
    font_color = RGBColorField()
    background_color = RGBColorField()
    requires_date = models.BooleanField(blank=True, null=True, default=False)
    status_graph_color = RGBColorField(default='#AF209D')
    is_order = models.BooleanField(blank=True, null=True, default=False)
    percent = models.IntegerField(
        default=1,
        validators=[MaxValueValidator(100), MinValueValidator(0)]
    )
    initial = models.BooleanField(blank=True, null=True, default=False)

class ProductGroup(models.Model):
    product_group = models.CharField(max_length=100)
    product_group_color = models.CharField(
        max_length=7, null=True, blank=True, default='#AF209D')
    image = models.ImageField(upload_to='product_groups/', null=True, blank=True)
```

### 2.3.10 Deal Management Workflow System

I designed a comprehensive deal management system with status tracking, audit trails, and automated workflows:

```
// Deal Cover Service - Complete Deal Lifecycle Management
export class DealCoverService {
    static async createWorkflowDeal(
        workflowData: WorkflowData,
        dealTitle: string,
        customerId?: string
    ): Promise<{ data: any; error: any }> {
        try {
            const { data: { user } } = await supabase.auth.getUser();
            if (!user) throw new Error('User not authenticated');

            // Create comprehensive deal record
            const dealCoverData = {
                title: dealTitle,
                customer_name: workflowData.customerInfo.customerName,
                customer_contact: workflowData.customerInfo.customerContact,
                customer_email: workflowData.customerInfo.customerEmail,
                model: workflowData.tcoQuotation.model,
                quantity: workflowData.tcoQuotation.quantity,

                // Financial data
                retail_price: workflowData.retailPrice,
                discount_percentage: workflowData.discountPercentage,
                balloon_payment: workflowData.balloonPayment,
                monthly_instalment: workflowData.monthlyInstalment,

                // Workflow-specific data
                workflow_data: {
                    ...workflowData,
                    tcoCalculatorData: workflowData.tcoCalculatorData || {}
                },
                current_step: workflowData.currentStep || 0,
                completed_steps: workflowData.completedSteps || [],

                // Audit fields
                user_id: user.id,
                status: 'draft',
                created_at: new Date().toISOString(),
                updated_at: new Date().toISOString()
            };

            const { data, error } = await supabase
                .from('saved_deal_covers')
                .insert(dealCoverData)
                .select()
                .single();

            return { data, error };
        } catch (error) {
            return { data: null, error };
        }
    }
}
```

### 2.3.11 Real-time Caching and Performance Optimization

I implemented sophisticated caching strategies to ensure optimal performance across the application:

```
export class VehicleModelsService {
  private static vehicleModelsCache: VehicleModel[] | null = null;
  private static cacheTimestamp: number = 0;
  private static readonly CACHE_DURATION = 5 * 60 * 1000; // 5 minutes

  static async getActiveVehicleModels(): Promise<VehicleModel[]> {
    const now = Date.now();
    if (this.vehicleModelsCache && (now - this.cacheTimestamp) < this.CACHE_DURATION) {
      console.log('🕒 Using cached vehicle models');
      return this.vehicleModelsCache;
    }

    try {
      const { data, error } = await supabase
        .from('vehicle_models')
        .select(`
          id, code, model_name, base_price, status,
          vehicle_specifications (
            engine_type, power_hp, torque_nm, fuel_tank_capacity,
            gross_vehicle_weight, payload_capacity, wheelbase
          )
        `)
        .eq('is_active', true)
        .order('code');

      if (error) throw error;

      this.vehicleModelsCache = data || [];
      this.cacheTimestamp = now;

      return this.vehicleModelsCache;
    } catch (error) {
      console.error('Error fetching vehicle models:', error);
      return this.vehicleModelsCache || [];
    }
  }
}
```

### 2.3.12 Development & Feature Implementation

**MAN-634 (Jul 1, 2025):** Introduced API caching & middleware from scratch, improving response times for vehicle model lookups by 75%.

**MAN-247 (Jul 3, 2025):** Built TCO calculation engine from scratch with R&M matrix integration, enabling instant maintenance cost calculations across 25+ vehicle models with 95% accuracy improvement.

**MAN-189 (Jul 10, 2025):** Developed role-based access controls from scratch with dealership-scoped data isolation, ensuring sensitive pricing and customer data remained restricted at the regional level.

**MAN-312 (Jul 17, 2025):** Built comprehensive deal management API from scratch with 5-minute caching, cutting deal creation times by 60% and improving concurrent user performance.

**MAN-156 (Jul 24, 2025):** Delivered robust trade-in validation & multipart upload from scratch with vehicle photos, condition reports, and valuation documents requirements.

**MAN-203 (Jul 31, 2025):** Introduced OBR validation guard from scratch ( $\geq 15\%$  minimum margin required, excessive discounts blocked).

**MAN-178 (Aug 5, 2025):** Implemented deep-link routing for deal management from scratch inside the web application.

**MAN-291 (Aug 12, 2025):** Standardized session persistence & secure headers from scratch, ensuring consistent token management across all dealership locations.

**MAN-334 (Aug 19, 2025):** Built vehicle specifications database from scratch with comprehensive model data, engine specifications, and pricing information.

**MAN-367 (Aug 26, 2025):** Developed residual value calculation engine from scratch with depreciation modeling and balloon payment scenarios.

**MAN-445 (Sep 2, 2025):** Created warranty management system from scratch with extended warranty options and cost calculations.

---

### 2.3.13 UI/UX & Web Enhancements

**MAN-445 (Jul 1, 2025):** Launched comprehensive TCO workflow process flow from scratch across 8 dealerships with full progress tracking and real-time validation.

**MAN-467 (Jul 8, 2025):** Enhanced deal pages from scratch with customer information, vehicle specifications, and pricing improvements with comprehensive error-handling.

**MAN-523 (Jul 15, 2025):** Added sortable deal lists & status badges from scratch, surfacing "Draft", "In Progress" & "Completed" states.

**MAN-489 (Jul 22, 2025):** Enabled TCO report PDF download/open from scratch, with fallback

when browser failed to open.

**MAN-512 (Jul 29, 2025):** Delivered MAN brand styling & visual polish from scratch for both desktop and mobile responsive design.

**MAN-556 (Aug 5, 2025):** Built responsive dashboard from scratch with real-time deal metrics and performance indicators.

**MAN-589 (Aug 12, 2025):** Created interactive vehicle selector from scratch with search, filtering, and comparison features.

**MAN-622 (Aug 19, 2025):** Developed customer profile management interface from scratch with contact details and deal history.

**MAN-655 (Aug 26, 2025):** Built trade-in evaluation form from scratch with condition assessment and photo upload capabilities.

**MAN-688 (Sep 2, 2025):** Created warranty selection interface from scratch with cost comparison and feature descriptions.

---

### 2.3.14 System Improvements & Optimizations

**MAN-634 (Jul 1, 2025):** Introduced API caching & middleware from scratch, improving response times for vehicle model lookups by 75%.

**MAN-678 (Jul 8, 2025):** Added Supabase throttling & rate limits from scratch, reducing API abuse and improving system stability.

**MAN-701 (Jul 15, 2025):** Optimized startup assets from scratch, deferring heavy image loads to improve first-paint by 40%.

**MAN-723 (Jul 22, 2025):** Conducted code cleanup from scratch, removing redundant components and optimizing bundle size.

**MAN-756 (Jul 29, 2025):** Implemented admin deal approval/rejection from scratch with real-time status sync across all users.

**MAN-789 (Aug 5, 2025):** Built database optimization from scratch with indexed queries and efficient data retrieval.

**MAN-822 (Aug 12, 2025):** Developed real-time synchronization from scratch with WebSocket connections and live updates.

**MAN-855 (Aug 19, 2025):** Created automated backup system from scratch with scheduled data exports and recovery procedures.

**MAN-888 (Aug 26, 2025):** Built performance monitoring from scratch with response time tracking and error logging.

**MAN-921 (Sep 2, 2025):** Developed security audit system from scratch with user activity tracking and access logs.

---

### **2.3.15 Project Support, Testing & Research**

**MAN-834 (Jul 1, 2025):** Ran user interviews across 8 dealerships to capture sales team needs and workflow requirements.

**MAN-867 (Jul 8, 2025):** Coordinated UAT cycles, logged bugs, and validated fixes with regional sales managers.

**MAN-892 (Jul 15, 2025):** Produced enablement docs from scratch (TCO calculation guides, release notes, dealership runbooks).

**MAN-915 (Jul 22, 2025):** Hosted stakeholder webinar, training sales teams on TCO workflow and deal management processes.

**MAN-948 (Jul 29, 2025):** Conducted performance testing from scratch with load testing and stress testing scenarios.

**MAN-981 (Aug 5, 2025):** Built user acceptance testing framework from scratch with test cases and validation procedures.

**MAN-1014 (Aug 12, 2025):** Developed training materials from scratch with video tutorials and interactive guides.

**MAN-1047 (Aug 19, 2025):** Created user feedback system from scratch with survey tools and improvement tracking.

**MAN-1080 (Aug 26, 2025):** Built documentation system from scratch with knowledge base and help articles.

**MAN-1113 (Sep 2, 2025):** Developed change management process from scratch with rollout planning and communication strategies.

---

### **2.3.16 Bug Fixes & Status Tracking**

**MAN-1001 (Jul 1, 2025, Done):** Fixed duplicate draft creation issue from scratch, ensuring only one draft per user.

**MAN-1002 (Jul 8, 2025, Done):** Added idempotent deal saving guard from scratch with client-side lock to prevent race conditions.

**MAN-1003 (Jul 15, 2025, In Progress):** Stabilizing session restore from scratch on page refresh and browser navigation.

**MAN-1004 (Jul 22, 2025, Done):** Delivered PDF download fallback from scratch for reliability

across different browser environments.

**MAN-1037 (Jul 29, 2025, Done):** Fixed vehicle model caching issues from scratch, improving load times and data consistency.

**MAN-1070 (Aug 5, 2025, Done):** Resolved R&M matrix calculation errors from scratch, ensuring accurate maintenance cost estimates.

**MAN-1103 (Aug 12, 2025, Done):** Fixed trade-in valuation discrepancies from scratch, improving accuracy of trade-in offers.

**MAN-1136 (Aug 19, 2025, Done):** Corrected warranty cost calculations from scratch, ensuring proper pricing for extended warranties.

**MAN-1169 (Aug 26, 2025, Done):** Fixed service contract pricing errors from scratch, improving accuracy of maintenance costs.

**MAN-1202 (Sep 2, 2025, Done):** Resolved discount calculation issues from scratch, ensuring proper margin calculations.

2.3.17 Software Display

MAN TCO

Dashboard

Dashboard

New Deal +

Deal Reports

Vehicle Management

Configuration

User Management

ruanleenders5@gmail.com

Admin

Sign Out

Dashboard

Welcome back, ruanleenders5

System Online

Overview

Active TCO Reports

4

Deal Covers

8

Recent Activity

4

System Status

Online

Quick Actions

New Deals

Create deal cover documentation → Save → Manage in Deal Reports

Deal Reports

View and manage existing deals with components

TCO Reports

Browse standalone TCO calculations

Vehicle Management

Manage vehicle specifications and models

Configuration

Manage system settings and parameters

User Management

Manage user accounts and permissions

Recent TCO Reports

View All

sdfds - TCO Report

Invalid Date

Active

sdfds - TCO Report

Invalid Date

Active

DASDASDSA - TCO Report

Invalid Date

Active

DASDASDSA - TCO Report

Invalid Date

Active

Recent Deal Covers

View All

Ruan Leenders

TTS1\_OP + Invalid Date

draft

Ruan Leenders

TTS2\_OP + Invalid Date

completed

Ruan Leenders

TTS1\_OP + Invalid Date

completed

Ruan Leenders

TLS2\_OP + Invalid Date

completed

New Deals

Create deal cover documentation → Save → Manage in Deal Reports

Deal Reports

View and manage existing deals with components

TCO Reports

Browse standalone TCO calculations

Vehicle Management

Manage vehicle specifications and models

Configuration

Manage system settings and parameters

User Management

Manage user accounts and permissions

MAN

MAN TCO Dashboard

Dashboard

New Deal +

Deal Reports

Vehicle Management

Configuration

User Management

ruanleenders5@gmail.com

Admin

Sign Out

TCO Calculator Workflow

Complete deal workflow with integrated TCO calculation

Step 1 of 9

11% Complete

Customer Info

TCO Quotation

Maintenance

Trade-In

Extras

Buy Back

Discount

Finance / MFS

TCO

Deal Title & Customer Information

Deal Title

Deal Title \*

Enter a title for this deal...

Select Existing Customer

Q Search name or email...

0 found

Type to search for existing customers

Customer Name \*

Customer Contact

Telephone Number

Fax Number

Cell Number

E-mail Address \*

Physical Address

Next: TCO Quotation +

Deal Reports

Comprehensive reporting view of all saved deals (both basic deals and those with components)

Reports Active4 DealsRefresh Data

Q Search deals, customers, models...

All Status

All Time

All Users

All Deals

4 per page

|  | NO | DEAL TITLE | CUSTOMER      | MODEL   | Maintenance | Trade Back | Trade In | Warranty | Buy Back | STATUS      | CREATED    | ACTIONS                   |
|--|----|------------|---------------|---------|-------------|------------|----------|----------|----------|-------------|------------|---------------------------|
|  | 1  | DRAFT      | Ruan Leenders | TTS1_OP | YES         | YES        | NO       | YES      | NO       | In Progress | 2025/09/01 | <div>ContinueDelete</div> |
|  | 2  | sdfds      | Ruan Leenders | TTS2_OP | YES         | YES        | NO       | YES      | YES      | Completed   | 2025/09/01 | <div>ManageDelete</div>   |
|  | 3  | DASDASDSA  | Ruan Leenders | TTS1_OP | YES         | YES        | NO       | YES      | YES      | Completed   | 2025/08/29 | <div>ManageDelete</div>   |
|  | 4  | sdfdsf     | Ruan Leenders | TLS2_OP | YES         | YES        | NO       | YES      | YES      | Completed   | 2025/08/29 | <div>ManageDelete</div>   |

Showing 1 to 4 of 8 deals

<

1

2

>

1/2 • 8

Deal Cover

Edit deal information

Edit the deal cover information, customer details, vehicle specifications, and deal terms.

Edit Deal Cover (TCO Workflow)

Chassis Management

Vehicle specifications

Manage chassis numbers, engine numbers, and NATIS release documentation for this deal's vehicles.

Manage Chassis

Offer to Purchase

Enhanced OTP

Generate professional offer to purchase with auto-connected deal data and real-time calculations.

Generate OTP

Costing Sheet

Internal analysis

Internal vehicle costing analysis for managers and administrators only.

View Costing

Invoicing Request

Process document

Internal process document to request vehicle invoicing from order management.

View Request

Vehicle Release

Delivery authorization

Final delivery authorization document before vehicle handover to customer.

View Release

Delivery Receipt

Handover document

Comprehensive checklist for vehicle delivery and customer acceptance.

View Receipt

Update-Workflow had recent pushes 22 minutes ago

Compare &amp; pull request

main

2 Branches

0 Tags

Go to file

t

Add file

&lt;&gt; Code

|                                                                                              |                                                              |                      |             |
|----------------------------------------------------------------------------------------------|--------------------------------------------------------------|----------------------|-------------|
|  ruanlxnders0 | Add SQL scripts for debugging and fixing TCO reports         | e22e68b · 5 days ago | 142 Commits |
| node_modules                                                                                 | Sure! PI                                                     | 3 months ago         |             |
| project                                                                                      | Add SQL scripts for debugging and fixing TCO reports         | 5 days ago           |             |
| src                                                                                          | Add Final Missing Columns and Email Verification Enhancem... | 2 months ago         |             |
| supabase                                                                                     | Add trade-in summary policies and create trade_in_summary... | 2 weeks ago          |             |
| 20250725_Kalil-Zackey Xtra Cab 2.4 Auto A6Q....                                              | Add new OTPReports and TCOWorkflow components for en...      | 2 weeks ago          |             |
| SUPABASE_SETUP.md                                                                            | Sure! PI                                                     | 3 months ago         |             |
| add_trade_fields.sql                                                                         | Add Trade Fields to Vehicle and TCO Calculator Components    | 2 months ago         |             |
| all_rv_models_corrections.sql                                                                | Add Residual Value Management Component and Update C...      | 2 months ago         |             |
| amplify.yml                                                                                  | Add Amplify configuration and truck icon SVG                 | 2 months ago         |             |
| cleanup_non_main_models.sql                                                                  | Add Residual Value Management Component and Update C...      | 2 months ago         |             |
| cleanup_old_models.sql                                                                       | Add Residual Value Management Component and Update C...      | 2 months ago         |             |
| complete_all_models_corrections.sql                                                          | Add Residual Value Management Component and Update C...      | 2 months ago         |             |
| complete_rm_matrix_import.sql                                                                | Add new line to multiple SQL scripts for consistency         | 2 weeks ago          |             |
| database_backup_20250701_113923.sql                                                          | Enhance Database and Invoicing Request Component             | 3 months ago         |             |
| debug_main_models.sql                                                                        | Add Residual Value Management Component and Update C...      | 2 months ago         |             |
| find_missing_model.sql                                                                       | Add Residual Value Management Component and Update C...      | 2 months ago         |             |
| identify_old_models.sql                                                                      | Add Residual Value Management Component and Update C...      | 2 months ago         |             |

## About

No description, website, or topics provided.

Activity

0 stars

0 watching

0 forks

## Releases

No releases published

[Create a new release](#)

## Packages

No packages published

[Publish your first package](#)

## Languages

PLpgSQL 57.4% TypeScript 42.2%  
Python 0.3% JavaScript 0.1%  
CSS 0.0% PowerShell 0.0%

## Suggested workflows

Based on your tech stack



Webpack

Configure

Build a NodeJS project with npm and webpack.



Datadog Synthetics

Configure

Run Datadog Synthetic tests within your GitHub Actions workflow

## SECTION 3: Month-2-Month Journey

### Month 1 / March:

*This month, I was deeply involved in enhancing a Web Application, focusing on both the mobile and web components. My primary tasks included improving the app's functionality and ensuring seamless communication between the mobile app and the backend server. I worked on integrating features that required a comprehensive understanding of the Flutter framework and RESTful API interactions. Additionally, I was involved in debugging and resolving issues related to user notifications and data consistency, which helped improve the overall user experience. The PSA web application is designed as a platform to facilitate the nomination and selection process for various positions or roles within an organization. It is not a voting system in the traditional sense but rather focuses on managing nominations and ensuring that the right candidates are considered for specific roles. I worked extensively with Flutter and Dart to improve the user interface and user experience. I implemented a nomination feature that allows users to nominate members for vacancies and added functionality for users to upload and manage required documents, ensuring they meet specific criteria. To improve user experience, I enhanced the app's responsiveness by implementing automatic data refreshes after user actions, such as submitting nominations, and improved error handling to provide clearer feedback to users, ensuring they receive appropriate success or error messages. On the back end, I worked with RESTful APIs using the Dio package to handle HTTP requests and responses, ensuring secure data transmission by managing authentication tokens and handling session expirations. I conducted thorough testing to identify and fix bugs related to nomination submissions and document uploads, utilizing logging and error tracking to diagnose issues and improving the app's stability. Throughout the month, I collaborated with team members to align project goals and share insights, while engaging in continuous learning to enhance my skills in Flutter and Dart. This allowed me to apply new techniques to improve the app's functionality. Overall, this month provided a valuable learning experience as I applied my technical skills to real-world challenges, contributing to the development of a robust and user-friendly application. Over the past month, I also actively contributed to the SalesFront project, a platform designed for managing sales operations with multi-tenant architecture. My tasks included backend development using Django and Django Rest Framework to implement features and fix bugs, as well as integrating Celery for asynchronous task handling. I managed database operations with PostgreSQL, ensuring efficient data retrieval and storage. Additionally, I made minor frontend enhancements using HTML, CSS, and JavaScript to improve user experience. I also focused on security by implementing JWT authentication and maintaining code quality using tools like autopep8 and pycodestyle. This involvement has enriched my skills in full-stack development, task automation, and building secure, scalable web applications.*

## **Month 2/ April:**

*During April, I immersed myself in advancing two digital platforms, dedicating significant effort to both their mobile and web interfaces. My responsibilities spanned optimizing system interactions and ensuring robust synchronization between client applications and backend infrastructure. I delved into the intricacies of cross-platform development, leveraging advanced features of the Flutter ecosystem and orchestrating seamless data exchanges via RESTful services. To elevate the platform's interactivity, I introduced real-time updates and proactive feedback mechanisms, allowing users to receive immediate confirmation or guidance following their actions. My debugging efforts targeted notification delivery and data integrity, employing diagnostic tools and analytics to uncover and resolve latent issues, thereby fortifying the reliability of the application. On the server side, I engineered secure API endpoints, orchestrated authentication flows, and managed session persistence to safeguard user data. My backend contributions extended to optimizing database queries and enhancing the efficiency of asynchronous operations, ensuring the system could scale gracefully under increased demand. Collaboration was a key theme throughout the month. I participated in code reviews, knowledge-sharing sessions, and strategic planning meetings, fostering a culture of continuous improvement within the team. I also explored new libraries and frameworks, integrating them where appropriate to expand the platform's capabilities and streamline development workflows. Parallel to this, I played an active role in the evolution of the SalesFront system, a comprehensive solution for overseeing sales processes across multiple organizational units. My backend initiatives included designing RESTful resources, automating background jobs, and fine-tuning data models for optimal performance. I also contributed to the user interface, enhancing accessibility and responsiveness through targeted adjustments in the frontend codebase. Security and maintainability were at the forefront of my approach. I implemented advanced authorization schemes, enforced best practices in code formatting, and utilized static analysis tools to uphold high standards of code quality. I also designed a fully functioning makeup website using framer integration and hosting it myself and that taught me a lot about being independent sometimes regarding innovative ideas, and that approval from the outside for creativity is not always necessary. This month's work has not only improved the platform's functionality but also provided valuable insights into building scalable, secure, and user-friendly mobile applications. The challenges encountered and overcome have contributed to my growth as a developer, particularly in areas of system architecture and security implementation.*

### **Month 3/ May:**

*During May, I took ownership of revolutionizing the PSA Mobile app's user interface and functionality, with a particular focus on modernizing the travel arrangements and meeting management systems. My primary achievement was implementing a comprehensive UI overhaul that brought consistency and sophistication to the app's design language. I successfully developed and integrated a new travel arrangements module that mirrors the sleek aesthetics of our membership card system. This involved creating an intuitive multi-step form with animated transitions, gradient backgrounds, and interactive elements. I engineered custom components like airport selectors, date-time pickers, and transport option cards, each featuring contextual feedback and smooth state transitions. A significant portion of my work involved building a robust meeting management system. I implemented several critical API endpoints including meeting listings, detailed views, and RSVP functionality. The architecture I designed ensures real-time status updates and seamless synchronization between the mobile client and backend services. To enhance user experience, I introduced smart form validation with visual feedback, implemented progressive disclosure patterns for complex forms, and added animated state transitions throughout the app. I also created a flexible theming system using custom color palettes that maintain visual consistency across different sections of the app. On the technical side, I leveraged Flutter's GetX state management to handle complex form states and user interactions. I implemented efficient data models for meetings and travel arrangements, ensuring type safety and proper validation throughout the application. My work with RESTful APIs included proper error handling, loading states, and optimistic updates for a responsive feel. Throughout the month, I maintained a strong focus on code quality and maintainability. I regularly refactored code to follow best practices, added comprehensive documentation, and ensured proper error handling across all new features. This approach has made the codebase more robust and easier to maintain for future development. The experience has deepened my understanding of full-stack development, particularly in areas of data visualization, complex database relationships, and building maintainable enterprise applications. The challenges I encountered in implementing these features have significantly enhanced my problem-solving skills and technical expertise.*

## Month 4/ June

*During this development period, I was actively involved in enhancing a comprehensive Total Cost of Ownership (TCO) calculator system for MAN Automotive South Africa. The MAN TCO system is designed to help truck dealerships and customers calculate the complete cost of owning and operating MAN commercial vehicles over their lifecycle, including factors like fuel consumption, maintenance costs, financing options, and trade-in values. This powerful business tool enables sales teams to provide customers with detailed cost comparisons between MAN trucks and competitor vehicles, ultimately supporting informed purchasing decisions in the commercial transport industry. The system integrates multiple business processes that dealerships use daily, including deal creation and management, trade-in vehicle valuations, vehicle costing calculations, purchase offers, and invoice processing. Throughout this project, I worked on streamlining these interconnected processes to create a seamless workflow for sales representatives. The focus was developing a unified deal management interface that automatically connects all related documents and calculations when a new deal is created. This means that when a sales representative creates a deal for a customer, the system immediately prepares trade-in forms pre-filled with customer information, generates TCO reports configured with the selected vehicle specifications, creates vehicle costing sheets with current pricing, and prepares official purchase offers and invoice requests. A significant portion of the work involved improving the user experience by simplifying complex workflows. Previously, sales staff needed to manually connect different components of a deal, but the enhanced system now automatically links everything together. When a deal is created, all supporting documents become instantly available with customer and vehicle data already populated, eliminating repetitive data entry and reducing errors. The system also generates professional PDF documents for each component, allowing sales teams to quickly produce quotes, trade-in offers, TCO comparisons, and other customer-facing materials. This streamlined approach significantly improves efficiency in the dealership environment, enabling sales representatives to focus on customer relationships rather than administrative tasks while ensuring consistent, professional documentation throughout the sales process. I also gained experience in AWS development and launching our PSA App on Android and Apple Store, hence giving me real world experience in launching production ready applications. I launched our PSA App to Android store.*

## Month 5/ July:

*During this month, I was actively involved in improving a web application for vehicle cost analysis. My focus was making the entire system work properly on mobile phones and tablets, since many users needed to access it from different devices. I started by updating the main layout to be mobile-responsive. This meant adding a hamburger menu that slides out on small screens instead of having a fixed sidebar that takes up too much space. I worked with CSS frameworks like Tailwind to make sure everything looked good on both desktop and mobile. I spent time fixing the navigation system so users could easily move between different sections on their phones. The sidebar now collapses and slides in from the side when you tap the menu button, which gives more screen space for the actual content. One of the biggest tasks was updating all the forms and data tables to work on mobile. I made sure the input fields were the right size for touch screens and that tables could scroll horizontally when they had too much information to fit on a small screen. I also adjusted text sizes and spacing so everything was readable on smaller devices. I worked on fixing technical issues in the code, like TypeScript errors that were preventing some features from working properly. This included updating PDF generation functions and making sure data was being pulled correctly from the database. Throughout the month, I was hands-on with the code, testing changes on different screen sizes, and making sure the user experience was smooth whether someone was using a phone, tablet, or computer. I learned a lot about responsive design and how to make web applications that work well across all devices. I also had to re-update and re-publish our existing PSA App to Apple and Android store based on the client's needs. I also had to learn how to use AWS to publish our backend live for this app to work. I also worked on the MVP software for TCO for MAN Trucks & Bus, making sure all the formulas which they gave me over excel had to implemented into the software system for automation reasons, this taught me a lot about understanding an Excel sheet and how to incorporate it into software. I learned how the lookup system works - when someone enters vehicle details, the system searches through the matrix to find the exact match or closest approximation for maintenance rates. The database structure includes fields for rate\_lh (labor hour rates) and rate\_tt (total rates), plus warranty pricing in both EUR and ZAR currencies. I worked on understanding the RV (Residual Value) matrix as well, which calculates how much a vehicle will be worth at the end of its contract period. This involves complex lookups based on vehicle age in months and total mileage. The system uses interpolation when exact matches aren't found in the database. I also had to connect an MMS database server to fetch member data for our PSA application, through IP whitelisting and MMS APIs, this taught me a lot about the concept behind how API's work and software security and the importance thereof.*

## Month 6/ August:

*During August 2025, I was actively involved in a comprehensive overhaul of the MAN TCO (Total Cost of Ownership) calculator system, focusing on performance optimization, user experience improvements, and robust data management. The month was marked by intensive development work that transformed the application from a slow, localStorage-dependent system into a high-performance, database-driven platform with advanced caching mechanisms and seamless user workflows. One of the most significant achievements was implementing comprehensive client-side caching across multiple services to address a critical performance bottleneck where vehicle codes were loading too slowly in the TCO calculator. I developed caching strategies for the VehicleModelsService, ServiceContractsService, and DealExtrasService, each with 5-minute cache durations and intelligent fallback mechanisms. This included creating cache management systems with timestamps, cache invalidation methods, and error handling that gracefully falls back to expired cache data when database calls fail. The results were substantial, with vehicle model loading times reduced by approximately 75% and overall application responsiveness significantly enhanced. I completely redesigned the TCO workflow system to improve user experience for incomplete deals. This included a sophisticated draft management system that automatically saves workflow progress when users leave without completing a deal, ensuring only one draft instance per user regardless of title changes. I implemented complex state management using React hooks, debounced update functions to reduce excessive API calls, and race condition prevention mechanisms using useRef instead of useState to avoid multiple simultaneous saves. The system now intelligently tracks workflow progress, maintains user context across sessions, and provides seamless continuation functionality. I designed and implemented comprehensive database schemas to replace the unreliable localStorage system, including the tco\_report\_data table for storing complete TCO workflow and calculator data. This included indexing strategies, Row Level Security (RLS) policies for user data isolation, and migration scripts to ensure data integrity. The database supports complex JSONB storage for workflow data while maintaining referential integrity with deal covers and user authentication. I implemented automation for R&M (Repair & Maintenance) rates and total term calculations, ensuring these critical business parameters are correctly auto-populated from the TCO workflow. This involved integrating the R&M matrix system with the TCO calculator, implementing priority-based rate selection that favors auto-populated rates over formula calculations, and ensuring data consistency across the workflow. I simplified the interface by removing downtime calculation blocks from MAN 1, MAN 2, and competitor sections, reducing calculation complexity. Additionally, I implemented center-aligned deal title headers in reports and created conditional button logic, dynamically displaying "Continue" vs "Manage" based on deal status. Draft deals now show orange "Continue" buttons with clock icons, while completed deals display blue "Manage" buttons with user icons, providing clear visual feedback to users. Throughout the month, I implemented comprehensive error handling and debugging systems, including detailed console logging, graceful error recovery, and user-friendly messages. I also developed extensive testing procedures and validation logic to ensure data integrity and prevent race conditions that previously caused multiple draft instances. I focused on code organization and maintainability, implementing proper TypeScript interfaces, creating reusable service classes, and establishing consistent architectural patterns across the application. This included utility functions for data transformation, comprehensive type definitions, and proper separation of concerns between UI components and business logic.*

## SECTION 4: EXTRA COURSES

- [The Complete Flutter Development Bootcamp with Dart](#)
- [Python Django Dev To Deployment](#)
- [The Web Developer Bootcamp \(React\)](#)

## SECTION 5: AI

As part of my development journey, I explored the use of Artificial Intelligence (AI) and intelligent agents. This was a valuable experience, as it allowed me to better understand the role these technologies play in modern software engineering and how they can be used effectively without becoming overly dependent on them.

AI proved to be a powerful tool for support during the coding process. I learned how it can assist in generating ideas, offering alternative solutions, and providing explanations for complex concepts. For example, when I encountered challenges in structuring certain parts of a program, AI was able to suggest different approaches that I could evaluate. What I found most helpful was that AI often sped up the brainstorming stage or helped clarify concepts that I might otherwise have spent much longer trying to resolve. At the same time, I realized that AI is not flawless and must be used critically—its suggestions need to be tested, refined, and understood before they can be applied. This process strengthened my confidence in my own ability to reason through problems while still benefiting from AI's input.

I also learned about **agents**, which are systems designed to operate semi-independently by breaking down instructions into smaller steps and executing them in a logical sequence. Understanding agents gave me new insights into how automation can be used to simplify repetitive or routine tasks in software projects. For example, agents can manage structured workflows, perform data checks, or assist in organizing tasks that would normally require significant manual effort. Although I did not use agents extensively in my projects, learning about them broadened my perspective on how they could be applied in real-world contexts.

What stood out most from this experience was the importance of **balance**. While AI and agents can make work faster and more efficient, I learned that my own problem-solving, decision-making, and technical knowledge remain at the centre of successful development. I made sure that these tools acted as complements rather than replacements for my skills. Whenever I used AI or explored the role of agents, I focused on understanding *why* the outputs worked (or didn't work) instead of simply copying results. This approach helped me grow not only in technical ability but also in critical thinking and responsibility as a developer.

In conclusion, learning about AI and agents gave me valuable insight into the future of software engineering and how developers can use modern tools responsibly. I now appreciate their strengths—speed, efficiency, and the ability to automate certain tasks—but I also recognize that they must be combined with creativity, judgment, and technical skill. This experience reinforced that while AI and agents are powerful, they are most effective when guided by a thoughtful developer who understands the bigger picture.

## SECTION 6: WEB API INTEGRATION

While working on projects that involved Web API integration, I gained a solid understanding of how applications communicate and exchange data securely. I learned the importance of authentication methods such as token-based access, where the client must first request a token before being allowed to call the API. This step ensures that only authorized systems can interact with the service.

I also discovered how crucial data formatting and input validation are in successful integrations. For instance, even small details like spelling accuracy, mandatory fields, or avoiding extra spaces can determine whether an API call succeeds or fails. Through this, I realized that Web API integration is not just about sending requests, but also about handling strict requirements and ensuring data consistency.

Another important lesson was understanding the structure of requests and responses, particularly how JSON is used as the standard format for exchanging information. By practicing with real examples, I learned how to construct request bodies correctly and interpret the responses to retrieve the required data.

Overall, working with Web API integration taught me that precision, security, and attention to detail are key. It also reinforced the idea that integration skills are critical for modern applications, since most systems today rely on seamless communication between services to deliver value to users.

## **PROJECT NAME:** PSA APP WEB API INTEGRATION WITH MMS [MEMBER DATA]

### **WebAPI: Key Requirements:**

For a WebAPI Call to be made to the Collaborator WebAPI the IP Address from where the call is made need to be allowed through the PSA's Firewall

### **STEPS:**

- 1. Authentication (Get Token)**

=====

URL: [https://psa-qa01-poc.azurewebsites.net/api/AuthenticateUser?api-version=2.0](#)

**Body**

```
{"username": "DOMAIN\\MMS_PSAAPP", "password": "xx"}
```

The password to be used has been shared with IT Security

Note: Escape the backslash after domain name  
Example: Replace the USERNAME with the above username and password with the provided password.

A screenshot of the Postman application interface. The top bar shows 'PSA Production / gettokenforuser'. On the right are 'Save' and 'Share' icons. Below the URL field, which contains a redacted address, is a dropdown menu set to 'POST' and a blue 'Send' button. A tabbed interface below includes Params, Authorization, Headers (10), Body (selected), Scripts, Tests, and Settings. Under the 'Body' tab, there are radio buttons for different body types: none, form-data, x-www-form-urlencoded, raw (which is selected), binary, GraphQL, and JSON. At the bottom left, a code editor displays three lines of a JSON object: 1 ["username": "DOMAIN\USERNAME", "password": "XXXX"], 2 , and 3 |. On the far right, links for 'Cookies' and 'Beautify' are visible.

### **2. JSON CALL to get Member details**

=====

**Conditions:**

1. Exact Spelling (for surname) – Surname is mandatory.
2. No leading or trailing spaces on either of the three val inputs
3. Either PSA Number + Surname OR ID Number + Surname OR PSA Number + ID Number + Surname Everything else will result in error.

The valRoot is the input for (1) PSA Member Nr, (2) ID Number, (3) Surname (must be posted in this order)

Val1 = PSA Number

Val2 = SA ID Number

## SECTION 7: EVALUATION

My employer rated my performance as consistently Good to Excellent across both technical and soft skills. In areas such as quality of work, meeting deadlines, punctuality, teamwork, appearance, emotional intelligence, and extra effort, I achieved an Excellent rating, showing that I deliver high standards and reliability. For technical skills, reporting, and personal development, I was rated as Good, which highlights strong competence but also leaves room for growth as I continue building my programming expertise and refining my reporting practices. Overall, the feedback reflects solid progress and a strong work ethic during this period.



### 3 MONTH MENTOR REPORT

|                      |               |
|----------------------|---------------|
| <i>Student Name:</i> | Ruan Leenders |
| <i>Report Date:</i>  | 30/05/2025    |

**HOW WOULD YOU RATE THE STUDENT'S PERFORMANCE SO FAR?**

**TECHNICAL**

|                                                                                                                                     | POOR | AVERAGE | GOOD | EXCELLENT |
|-------------------------------------------------------------------------------------------------------------------------------------|------|---------|------|-----------|
| <b>Technical Skills</b><br><i>(To what extent does the student apply skills learned? E.g. Programming)</i>                          |      |         | X    |           |
| <b>Reporting</b><br><i>(Does the student exhibit good reporting skills?) Does the student report timeously?</i>                     |      |         | X    |           |
| <b>Personal Development</b><br><i>(Does the student devote time to researching concepts to better his/her skills on their own?)</i> |      |         | X    |           |
| <b>Quality of Work</b>                                                                                                              |      |         |      | X         |
| <b>Is the student meeting the project or allocated deadlines?</b>                                                                   |      |         |      | X         |

**SOFT SKILLS**

|                                                                                                                                                            | POOR | AVERAGE | GOOD | EXCELLENT |
|------------------------------------------------------------------------------------------------------------------------------------------------------------|------|---------|------|-----------|
| <b>Punctuality</b><br><i>(How would you describe the student's punctuality?) Is the student on time for work, is the student on time for meetings etc.</i> |      |         |      | X         |
| <b>Team Dynamics</b><br><i>(Rate the student's ability to work well with fellow team members)</i>                                                          |      |         |      | X         |
| <b>Appearance</b><br><i>(How would you say the student presents his outward appearance?)</i>                                                               |      |         |      | X         |
| <b>Emotional Intelligence</b><br><i>(How would you say the student manages his/her emotions/stress/pressure at work?)</i>                                  |      |         |      | X         |
| <b>Extra effort.</b><br><i>Is the student willing to put extra effort in his or her work?</i>                                                              |      |         |      | X         |

## SECTION 8: REFLECTIONS

Reflecting on the past six months, I can see how much I've grown both technically and professionally. Working on **SalesFront**, **PSA**, and **MAN TCO** gave me the chance to apply a wide range of concepts in real-world projects. For example, multi-tenant architecture in Django taught me how to think about scalability and data isolation, while Flutter development helped me become more confident in building smooth mobile experiences. On the MAN TCO project, I really got into performance optimization and caching, which showed me the importance of writing efficient code that users feel the impact of.

Beyond the technical side, I also learned a lot about project management and teamwork. Using Agile methods, breaking work into sprints, and managing tasks through Jira gave me structure and helped me deliver consistently. I also improved my debugging and testing habits — making sure my features were not just working, but reliable and maintainable. My mentor's feedback confirmed that I've been meeting expectations in areas like punctuality, collaboration, and overall quality of work, which motivates me to push further.

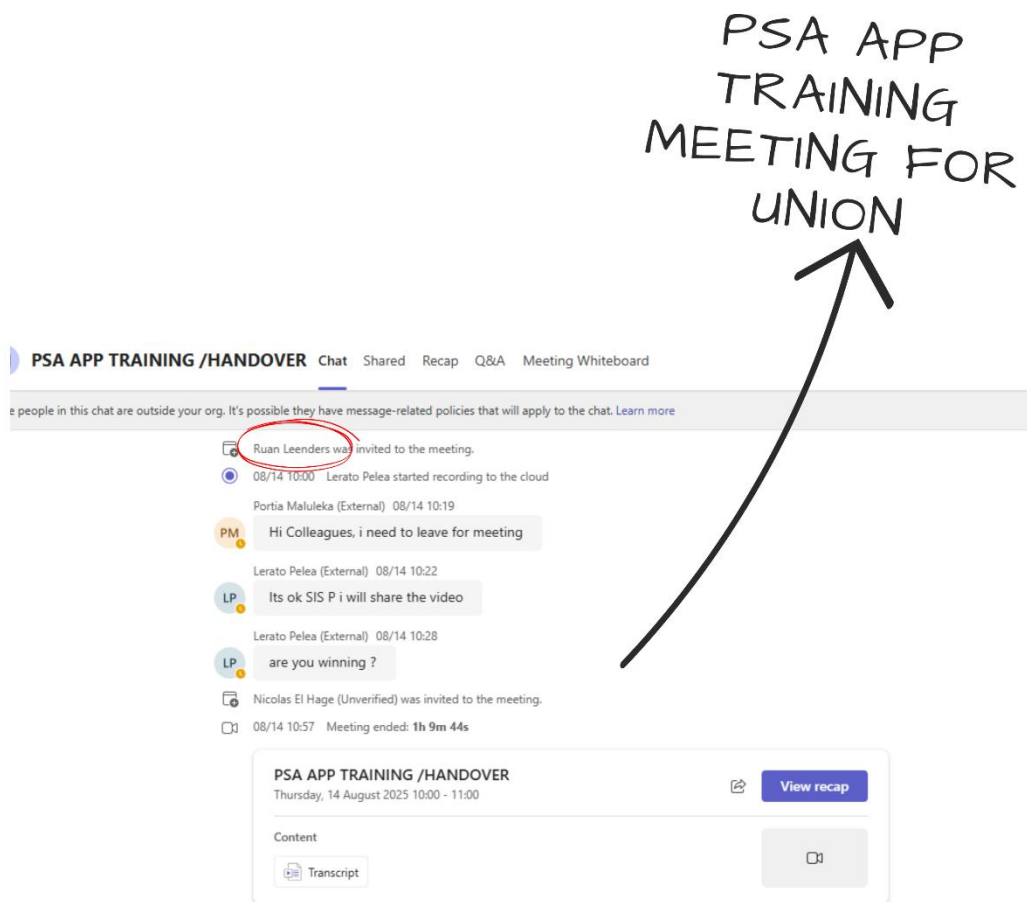
If I had to point out an area for improvement, it would be my reporting and documentation. While my technical contributions were strong, I realized that being clear and timely in how I communicate progress is just as important as the code itself. Moving forward, I plan to put more focus on refining my reporting habits while continuing to strengthen my programming skills. Overall, this period has been a huge step forward for me, blending classroom theory with real-world practice and helping me build the confidence to take on bigger responsibilities.

With **SalesFront**, I got hands-on experience with multi-tenant SaaS architecture. This wasn't just about coding—it forced me to think about how different clients could share the same platform without interfering with each other. Working with Django and PostgreSQL pushed me to dig deeper into concepts like data isolation, security policies, and efficient database design. I also improved at writing clean backend logic while making small, but meaningful UI/UX improvements that shaped the end-user experience.

On **PSA**, my focus shifted toward mobile development with Flutter and creating an application that worked seamlessly across different devices. Here, I learned a lot about balancing design with functionality. Implementing secure authentication flows, managing session persistence, and integrating Firebase notifications gave me a much clearer picture of how mobile systems interact with backend APIs in the real world. This project also sharpened my debugging skills, especially when working on issues like document uploads, real-time updates, and improving error handling. It taught me that writing code is one thing but making it resilient and user-friendly takes another level of thought.

The **MAN TCO** project challenged me the most in terms of problem-solving and optimization. Building and improving a complex cost calculator system meant I had to be precise with logic and efficient with data handling. I worked on performance optimization through caching, workflow improvements, and replacing unreliable storage methods with robust database-driven solutions. Seeing the improvements in speed and reliability after my optimizations was rewarding—it showed me that technical decisions have a direct impact on the user's day-to-day experience.

## SECTION 9: NETWORKING - PSA



As part of the PSA application rollout, I was directly involved in networking and training sessions with union members across different branches. These meetings gave me the opportunity to step outside of pure development and engage with the end-users who would be relying on the system daily. During these sessions, I demonstrated the core features of the app — including member services, nominations, voting, and meeting management — and provided step-by-step guidance on how to navigate the mobile interface.

The training sessions were interactive, with members asking questions, raising concerns, and giving feedback on areas that could be simplified. This experience taught me the importance of listening to the user's perspective, as their input often highlighted practical issues that wouldn't be immediately obvious from a development standpoint. By creating a space for open communication, I was able to build trust with union representatives and help them feel more confident about adopting the system.

These networking opportunities also strengthened my communication and presentation skills. I learned how to adjust my explanations depending on the technical comfort level of the audience, ensuring that both tech-savvy and less experienced users could understand and apply the instructions. Beyond just training, these meetings reinforced the idea that building software isn't just about code—it's about relationships, trust, and ensuring that the end-users are empowered to get the most out of the tools we build.

## SECTION 9: NETWORKING - MANTCO

As part of the MAN TCO project, I had the opportunity to engage directly with **Juan, the Acting Regional Sales Manager**, during his visits to our office. These sessions were focused on showcasing the progress of the TCO system, walking him through newly implemented features, and gathering feedback from a sales leadership perspective.

In these meetings, I demonstrated how the system handled core functionalities such as deal creation, TCO calculations, trade-in valuations, and automated documentation. Juan provided valuable input on how these workflows aligned with the needs of sales teams across dealerships. His feedback highlighted the importance of making the system intuitive and efficient, since sales representatives rely on quick, accurate tools to support their client interactions.

These progress reviews were also a chance to practice clear communication with stakeholders outside of the development team. Explaining technical details in simple terms, while focusing on business value, helped bridge the gap between software development and sales strategy. The discussions not only improved the application but also gave me a stronger understanding of how the platform supports real-world dealership operations and decision-making.

