```
+       # Check if database exists
+       result = await conn.execute(text(f"SELE
```

✕ Check failure

Code scanning / Semgrep OSS

Semgrep Finding: python.sqlalchemy.security.audit.avoid-sqlalchemy-text.avoid-sqlalchemy-text

Error

sqlalchemy.text passes the constructed SQL statement to the database mostly unchanged. This means that the usual SQL injection protections are not applied and this function is vulnerable to SQL injection if user input can reach here. Use normal SQLAlchemy operators (such as or_(), and_(), etc.) to construct SQL.

Show more details

Dismiss alert

Reply...

dylen-engine/scripts/init_db.py

```
+
+       if not exists:
+         print(f"Database '{target_db}' does n
+         await conn.execute(text(f'CREATE DATA
```

✕ Check failure

Code scanning / Semgrep OSS

Semgrep Finding: python.sqlalchemy.security.audit.avoid-sqlalchemy-text.avoid-sqlalchemy-text

Error

sqlalchemy.text passes the constructed SQL statement to the database mostly unchanged. This means that the usual SQL injection protections are not applied and this function is vulnerable to SQL injection if user input can reach here. Use normal SQLAlchemy operators (such as or_(), and_(), etc.) to construct SQL.

Show more details

Dismiss alert

Reply...

dylen-engine/app/api/routes/auth.py

```
+   try:
+     decoded_token = await run_in_threadpool(v
+   except Exception as e:
+     logger.error("Login failed: Token verific
```

⚠ Check warning

Code scanning / Semgrep OSS

Semgrep Finding: python.lang.security.audit.logging.logger-credential-leak.python-logger-credential-disclosure ⚠ Warning

Detected a python logger call with a potential hardcoded secret "Login failed: Token verification crashed: %s" being logged. This may lead to secret credentials being exposed. Make sure that the logger is not logging sensitive information.

Show more details

Dismiss alert

Reply...

dylen-engine/app/api/routes/auth.py

```
+   # Extract identity details for downstream l
+   firebase_uid = decoded_token.get("uid")
+   # Masked email for debug if needed, but pre
+   logger.debug("Token verified. UID: %s", fir
```

⚠ Check warning

Code scanning / Semgrep OSS

Semgrep Finding: python.lang.security.audit.logging.logger-credential-leak.python-logger-credential-disclosure ⚠ Warning

Detected a python logger call with a potential hardcoded secret "Token verified. UID: %s" being logged. This may lead to secret credentials being exposed. Make sure that the logger is not logging sensitive information.

Show more details

Dismiss alert

Reply...

dylen-engine/app/api/routes/auth.py

```
+   try:
+     decoded_token = await run_in_threadpool(v
+   except Exception as e:
+     logger.error("Signup failed: Token verifi
```

⚠ Check warning

Code scanning / Semgrep OSS

Semgrep Finding: python.lang.security.audit.logging.logger-credential-leak.python-logger-credential-disclosure ⚠ Warning

Detected a python logger call with a potential hardcoded secret

"Signup failed: Token verification crashed: %s" being logged. This may lead to secret credentials being exposed. Make sure that the logger is not logging sensitive information.

Show more details

Dismiss alert

Reply...

dylen-engine/app/api/routes/auth.py

```
+    firebase_uid = decoded_token.get("uid")
+    token_email = decoded_token.get("email")
+    provider_id = decoded_token.get("firebase",
+    logger.debug("Signup token verified. UID: %
```

⚠ Check warning

Code scanning / Semgrep OSS

Semgrep Finding: python.lang.security.audit.logging.logger-credential-leak.python-logger-credential-disclosure

⚠ Warning

Detected a python logger call with a potential hardcoded secret "Signup token verified. UID: %s, Provider: %s" being logged. This may lead to secret credentials being exposed. Make sure that the logger is not logging sensitive information.

Show more details

Dismiss alert

Reply...

dylen-engine/app/notifications/email_sender.py

```
+      )
+
+      try:
+        with urllib.request.urlopen(request, ti
```

⚠ Check warning

Code scanning / Semgrep OSS

Semgrep Finding: python.lang.security.audit.dynamic-urllib-use-detected.dynamic-urllib-use-detected

⚠ Warning

Detected a dynamic value being used with urllib. urllib supports 'file://' schemes, so a dynamic value controlled by a malicious actor may allow them to read arbitrary files. Audit uses of urllib calls to ensure user data cannot control the URLs, or consider using the 'requests' library instead.

Show more details

Dismiss alert

Reply...

dylen-engine/app/notifications/push_sender.py

```
+
+   def send(self, notification: PushNotificati
+     """Drop the notification while recording 
+     logger.debug("Push notifications disabled
```

Check warning

Code scanning / Semgrep OSS

Semgrep Finding: python.lang.security.audit.logging.logger-credential-leak.python-logger-credential-disclosure

Warning

Detected a python logger call with a potential hardcoded secret "Push notifications disabled; dropping push token_prefix=%s" being logged. This may lead to secret credentials being exposed. Make sure that the logger is not logging sensitive information.

Show more details

Dismiss alert

Reply...

dylen-engine/verify_headers.py

```
+
+   req.add_header("Origin", origin)
+
+   with urllib.request.urlopen(req) as respons
```

Check warning

Code scanning / Semgrep OSS

Semgrep Finding: python.lang.security.audit.dynamic-urllib-use-detected.dynamic-urllib-use-detected

Warning

Detected a dynamic value being used with urllib. urllib supports 'file://' schemes, so a dynamic value controlled by a malicious actor may allow them to read arbitrary files. Audit uses of urllib calls to ensure user data cannot control the URLs, or consider using the 'requests' library instead.

Show more details

Dismiss alert

Reply...

gemini-code-assist bot commented 1 hour ago now supports DB-backed runtime configuration and feature flags, offering granular scoping (GLOBAL, TIER, TENANT) and RBAC controls for dynamic application behavior management.

- **Firebase Integration for Authentication and RBAC**: Firebase is integrated for user authentication, with custom claims synchronized to enforce Role-Based Access Control (RBAC) across the application.
- **Comprehensive Documentation**: Extensive new documentation has been added, covering security policies, migration guides, GCP deployment, frontend integration, and a detailed code quality refactoring plan, ensuring clear guidelines for all aspects of the project.

▶ **Ignored Files**

▶ **Using Gemini Code Assist**

---

1. Review the Privacy Notices, Generative AI Prohibited Use Policy, Terms of Service, and learn how to configure Gemini Code Assist in GitHub here. Gemini can make mistakes, so double check it and use code with caution. ↩

gemini-code-assist bot reviewed 1 hour ago

View reviewed changes

gemini-code-assist bot left a comment

Code Review

This pull request introduces a significant number of changes, including a major refactoring and renaming from 'DGS' to 'Dylen'. The changes add robust development, security, and deployment configurations. My review has focused on critical security issues and inconsistencies in the new setup. The most critical issue is the inclusion of a private key and certificate in the repository, which must be removed. Other comments address hardcoded credentials in an example file, version mismatches, and inconsistencies between Docker, Docker Compose, and Makefiles. This PR also adds a large number of IDE-specific configuration files, which are generally better to exclude from version control to avoid conflicts between developers.

`.certs/origin.key` Outdated

Comment on lines 1 to 5

```
1 + -----BEGIN PRIVATE KEY-----
2 + MIGHAgEAMBMGByqGSM49AgEGCCqGSM49AwEHBG0wawIBA
3 + 1w+eBh7dRKhr1EQkTYhWbrHx0OWhRANCAASWc6oM9e4ph
4 + Z6xq4ujuzujVaDAYsdOI3Tjol796YR5anjr6CDzTrRJT7
5 + -----END PRIVATE KEY-----
```

gemini-code-assist bot 1 hour ago

Security Critical

Committing private keys and certificates directly into the repository is a significant security risk. Even if these are for development, they should never be in version control. The `.gitignore` file is already configured to ignore the `.certs/` directory, but these files were still added. Please remove these files from the repository's history and provide instructions for developers on how to generate or obtain them for local development.

Open **Dle integration** #40
andesec wants to merge 209 commits into `main` from `dle-integration`

`.env.example`

Comment on lines +53 to +54

```
53 + DATABASE_URL=postgresql://dylen:dylen_passwor
54 + DYLEN_PG_DSN=postgresql://dylen:dylen_passwor
```

gemini-code-assist bot 1 hour ago

Security High Priority

The example environment file contains hardcoded credentials (`dylen:dylen_password`). While this is an example file, it's a common source of security vulnerabilities when developers copy it for their local `.env` and forget to change the defaults. This is especially risky if the same password is used in other environments. It's recommended to use non-secret placeholders for credentials, for example: `postgresql://<user>:<password>@localhost:5432/dylen` .

```
DATABASE_URL=postgresql://<user>:<password>@localhost:5432/dylen
DYLEN_PG_DSN=postgresql://<user>:<password>@localhost:5432/dylen
```

Reply...

Resolve conversation

`.snyk`

```
+
+ # Language settings
+ language-settings:
+   python: '3.12'
```

gemini-code-assist bot 1 hour ago

Medium Priority

There is a version mismatch for Python. This file specifies version `3.12`, but the `.python-version` file specifies `3.13`. This could cause Snyk to analyze dependencies for the wrong Python version, potentially missing vulnerabilities or reporting false positives. Please ensure these versions are consistent.

```
  python: '3.13'
```

Reply...

Resolve conversation

`Dockerfile`

```
+ EXPOSE 8002
+
+ # Run the application.
+ CMD ["python", "dylen-engine/entrypoint.py"]
```

gemini-code-assist bot 1 hour ago

Medium Priority

The `CMD` for the `production` stage is set to run `dylen-engine/entrypoint.py`. However, the `docker-compose.yml` file overrides this to run `uvicorn` directly. This makes `entrypoint.py` effectively unused when running with Docker Compose and can be confusing. If `entrypoint.py` is not needed for Docker, consider aligning the `Dockerfile`'s `CMD` with the `docker-compose.yml` command for clarity and consistency.

Reply...

Resolve conversation

`Makefile`

```
+    uv export --format requirements-txt --no-
+    @# Remove editable install (-e .) and loc
+    @sed -i.bak '/^-e/d' requirements.txt && 
+    @sed -i.bak '/^myapplication/d' requireme
```

gemini-code-assist bot 1 hour ago

Medium Priority

The `security-sca` target contains a `sed` command that uses the placeholder `myapplication`. This will not work as intended and should be replaced with the correct package name for this project (e.g., `dylen-engine`) or removed if not applicable. Using a placeholder can cause this part of the security scan to be skipped silently.

```
uv export --format requirements-txt --no-hashe
-output-file requirements.txt
@# Remove editable install (-e .) and local
packages from requirements.txt to satisfy Snyk
@sed -i.bak '/^-e/d' requirements.txt && rm
requirements.txt.bak
@sed -i.bak '/^dylen-engine/d' requirements.txt
&& rm requirements.txt.bak
```

Reply...

Resolve conversation

chore: remove local postgres data and untrack local secrets ✕ 67626ae