

**Djanjo Web Project**  
**DayScript (A Personal Diary)**  
Personal Diary / Journal Web Application

Junaid Mohi Ud Din    Enrollment: 01-134222-071

Shayan Ikram Abbasi    Enrollment: 01-134222-140

Arif Hussain            Enrollment: 01-134222-029

December 2025

# Contents

<b>1</b>	<b>Project Overview</b>	<b>3</b>
<b>2</b>	<b>Objectives</b>	<b>3</b>
<b>3</b>	<b>Team</b>	<b>3</b>
<b>4</b>	<b>Technology Stack</b>	<b>3</b>
<b>5</b>	<b>System Architecture</b>	<b>3</b>
5.1	MVT Mapping . . . . .	3
5.2	Database Schema (SQLite) . . . . .	3
<b>6</b>	<b>Features Implemented</b>	<b>4</b>
<b>7</b>	<b>URLs</b>	<b>4</b>
<b>8</b>	<b>Setup and Execution</b>	<b>4</b>
<b>9</b>	<b>Testing Plan</b>	<b>4</b>
<b>10</b>	<b>Screenshots (placeholders)</b>	<b>4</b>
<b>11</b>	<b>Limitations and Future Work</b>	<b>4</b>
<b>12</b>	<b>Conclusion</b>	<b>5</b>

# 1 Project Overview

DayScript is a Django-based personal diary/journal web application designed to manage textual entries only. It demonstrates Django's Model-View-Template (MVT) architecture, user authentication, CRUD operations, templating, forms, admin interface, and SQLite storage. The project aligns with the Web Engineering course requirements.

## 2 Objectives

- Implement Django MVT in a practical text-only application.
- Provide user authentication (signup, login, logout).
- Enable CRUD for diary entries with moods and tags.
- Support search and filtering (mood, date range, tag).
- Use SQLite as the default database.
- Offer an admin interface for data management.

## 3 Team

- Junaid Mohi Ud Din (01-134222-071)
- Shayan Ikram Abbasi (01-134222-140)
- Arif Hussain (01-134222-029)

## 4 Technology Stack

- Python 3.12, Django 5.2.9
- SQLite (default)
- HTML5, CSS3 (custom styling), Django templates/forms

## 5 System Architecture

### 5.1 MVT Mapping

- **Models:** Tag, DiaryEntry
- **Views:** Class-based and function-based (home, detail, create, update, delete, profile, register)
- **Templates:** Base layout, home, entry detail, entry form, delete confirm, login, register, profile
- **URLs:** Project `dayscript/urls.py`, app `diary/urls.py`

### 5.2 Database Schema (SQLite)

- **Tag:** `id`, `name` (unique), `created_at`
- **DiaryEntry:** `id`, `user`, `title`, `content`, `mood`, `created_at`, `updated_at`, M2M tags

## 6 Features Implemented

<b>Authentication</b>	Signup, login, logout (Django auth)
<b>CRUD Entries</b>	Create, read, update, delete diary entries
<b>Moods</b>	Predefined mood choices with emoji display
<b>Tags</b>	Existing tag selection + add-new tags in form
<b>Search/Filter</b>	Text search; filters: mood, date range, tag
<b>Profile</b>	Stats: total entries, mood distribution, tags used, recent entries
<b>Admin</b>	Manage users, entries, tags via Django admin
<b>UI/UX</b>	Responsive, modern CSS theme; pagination for entries

## 7 URLs

- / (home/feed with search/filter)
- /register/, /login/, /logout/
- /entry/new/, /entry/<id>/, /entry/<id>/edit/, /entry/<id>/delete/
- /profile/
- /admin/

## 8 Setup and Execution

1. **Create venv (optional but recommended):** `python -m venv .venv`
2. **Activate venv (Windows):** `.venv\Scripts\Activate`
3. **Install deps:** `pip install -r requirements.txt`
4. **Migrate DB:** `python manage.py migrate`
5. **Create superuser:** `python manage.py createsuperuser`
6. **Run server:** `python manage.py runserver`
7. Open `http://127.0.0.1:8000/`

## 9 Testing Plan

- Register, login, logout.
- Create entries with mood + tags; edit; delete.
- Search and filter by mood/date/tag.
- Profile page statistics and recent entries.
- Admin: manage users, entries, tags.

## 10 Screenshots (placeholders)

- Home/feed
- Create/Edit entry form
- Profile page
- Admin entries/tags

## 11 Limitations and Future Work

- No rich text editing; plain text only (per requirement).
- No file uploads/media (per requirement).

- Future: add reminders, calendar view, exporting entries.

## 12 Conclusion

DayScript fulfills the course requirements: Django MVT, authentication, CRUD with forms and templates, admin management, SQLite storage, search/filtering, and responsive UI. It is ready for submission.