STUDENT FEEDBACK DJANGO WEB APPLICATION.

Thursday, August 10, 2023 12:19 PM

Table of contents.

1. Introduction.

The Student Feedback Django Web Application is a comprehensive tool designed to facilitate the collection, analysis, and reporting of feedback from students about various aspects of an educational institution. This documentation provides an in-depth overview of the web application, its architecture, features, installation steps, and usage guide.

2. System Overview.

- Architecture.

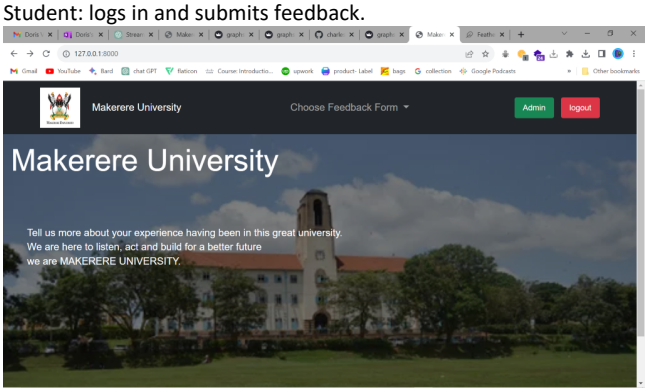
  The web application follows the model-view-controller(MVC) architecture pattern, with a strong emphasis on the Django framework's built-in components.
- Model: represents the data structure and interacts with the database.
- View: defines how data is presented to users.
- Controller: handles user input, manages data flow, and updates the model and view.

- Technologies used.

  Django framework: Provides the foundation for building the web application.
  Python: The primary programming language used for server-side logic.
  Google sheets: For developing the feedback questions.
  HTML, CSS, JavaScript: Used for front-end development and user interface.
  Spreadsheets: for storing feedback.
  Streamlit: Utilized for data visualization and analytics.

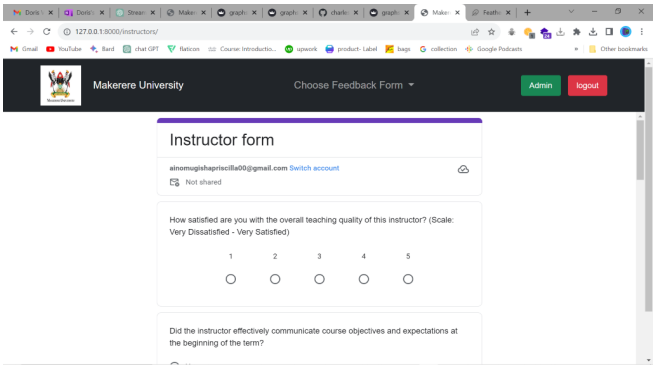
3. Features.

- User roles.
- Admin: manages user accounts, reviews feedback submissions, generates reports.
- Student: logs in and submits feedback.



The landing page.

- Feedback submission.
- Students can provide feedback on courses, instructors and campus facilities.
- Feedback includes ratings, comments and optional suggestions.



A sample of the feedback forms to be filled by the student.

- Data Management.
- All feedback submissions are stored in spreadsheets and google drive.
- Data integrity and security are maintained through Django's built-in mechanisms.

- Analytics and reporting.
- Admin users can generate reports and visualize trends using graphs.
- Reports provide insights into course and facility effectiveness, and instructor performance.

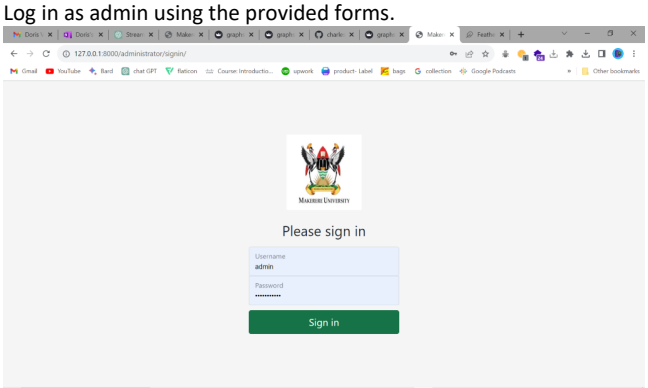
4. Installation and setup.

- Prerequisites
  - Python(3.11.1)
  - Django
  - Google sheets
  - Streamlit

- Installation steps
  - Clone the repository.
  - Install dependencies
  - Configure database settings
  - Run migrations
  - Create an admin user
  - Start the development server

5. Usage guide.

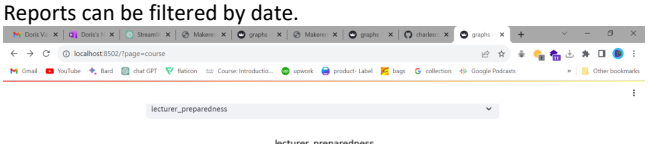
- Access the application via a web browser.
- Log in as admin using the provided forms.

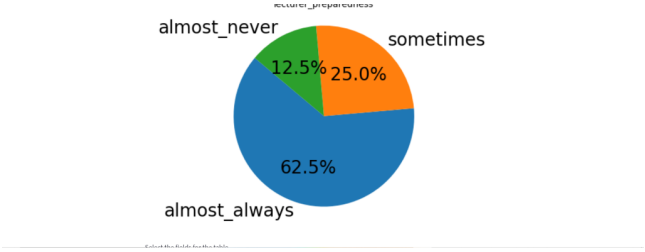


The log in page for administrator.

- Providing feedback.
- Students can provide rating and comments based on the experience.
- Optional suggestions can also be includes.

- Viewing feedback.
- This is done by the administrator.
- Graphs provide visual insights into feedback trends.
- Reports can be filtered by date.





A sample of data visualized by the administrator.

6. **Troubleshooting.**
- Ensure all prerequisites are properly installed.
- Check database configurations in 'settings.py'.
- Review error messages for guidance.

7. **Future enhancements.**
- Integration with email notifications for admins.
- Support for anonymous feedback submission.
- Multi-language support for international users.
- Integration with external tools for enhanced analytics.

8. **Conclusion.**
The student feedback Django we application serves as a valuable tool for educational institutions to gather and analyze feedback from students, facilitating continuous improvement in course quality and instructor performance.

9. **References.**
Django documentation: https://docs.djangoproject.com/
Google sheets Documentation: https://support.google.com/docs
Streamlit Documentation: https://docs.streamlit.io/