ADVISOR SURVEY APPLICATION

Improving How Advisors View and Access Survey Results

Dilraj Kaur

WWU Department of Computer Science

Walla Walla University
SEVENTH-DAY ADVENTIST HIGHER EDUCATION

ABSTRACT

This project involved developing a web application for the annually conducted Advisor Survey by the Academic Advisement Office to store and manage survey results. The application provides a centralized platform for securely storing and accessing results across multiple years. It features interactive dashboards that summarize the current year's responses and include comparative charts across advisors. Users log in through the university's Single Sign On system, with role-based access controls ensuring appropriate permissions. The solution significantly improves efficiency and accessibility by replacing a manual and fragmented distribution method with an organized, automated system.

INTRODUCTION

The Walla Walla University Academic Advisement Office sends out an Advisor Survey every year to collect feedback from students about their advising experience. In the past, staff compiled individual advisor PDF files and distributed them through email. Advisors then had to search through their inboxes to find past survey results, which made it difficult to track changes or access information quickly. This application addressed these challenges by introducing a secure, web-based platform for the Academic Advisement Office to store, manage, and analyze advisor survey results. The system includes a unified dashboard that allows users to view trends, compare data, and access historical records in an organized and user-friendly format.

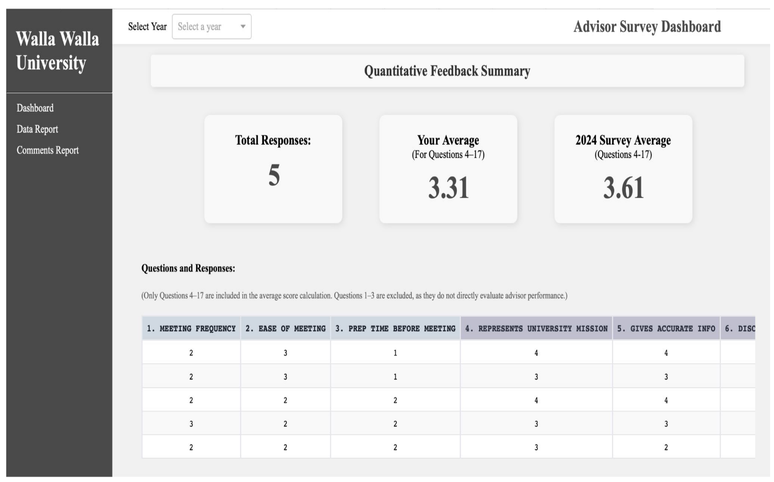

Fig. 1: Dashboard that shows key information and numeric values.

DEVELOPMENT & TESTING

Fig. 2: Tools used: HTML/CSS, Dash and Dokku

The web app was developed using a range of tools and frameworks. Dash in Python was used to build interactive charts and display key information such as averages, response counts, suggestions, and comparisons. HTML and CSS shaped the layout, while Dash callbacks enabled features like dropdown-based graph updates and year-based filtering to improve user experience.

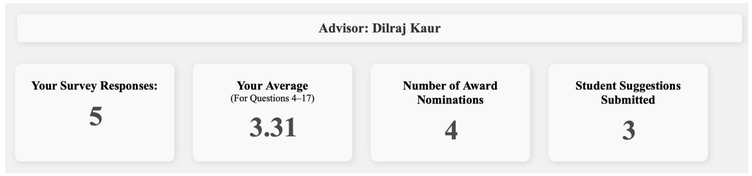

Fig. 3: Dashboard that shows key information and numeric values.

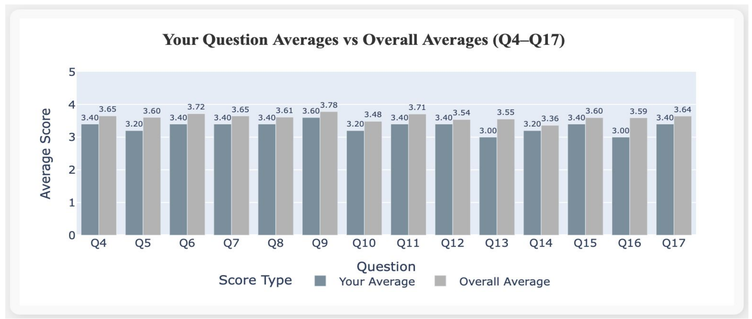

Fig. 4: A comparative chart created using Dash.

Flask handled server-side logic, and Authlib managed authentication via the university's Single Sign-On system. GitLab supported version control and CI/CD automation, and the app was deployed using Dokku with environment variables for secure configuration. ChatGPT was used as a development support tool to troubleshoot coding issues, interpret errors, and refine logic.

For testing, the pytest framework was used to verify the behavior and outputs of key functions. Unit tests were written for data processing, layout generation, and various charting functions, and verification was done to ensure that callbacks triggered correctly and that visual components rendered expected output. Pytest was chosen because of its simple syntax.

CHALLENGES

Throughout the development process, a few challenges were encountered. Since this application was built from scratch, one of the biggest hurdles was designing the layout and determining how to organize all the components effectively. The survey application's requirements included the admin view, advisor interface, and sections for both numerical and written feedback, all of which needed to be presented clearly and intuitively for different users. Setting up authentication also presented difficulties. The university's IT team was closely involved to resolve issues related to OIDC configuration, including setting the proper redirect URLs and securely managing the necessary credentials.

SUMMARY

This project involved building a web application to modernize how Walla Walla University's Academic Advisement Office collects and shares Advisor Survey results. The system includes a centralized dashboard with dynamic charts and visual tools to help advisors review feedback, compare scores, and analyze trends over time.

Developed using Dash and Python, the app supports secure login, role-specific views, and multi-year data access. The current version is deployed to the university's internal server and is actively being evaluated by academic leadership. Future development will focus on incorporating more advanced metrics, visual improvements, and extended support for older survey formats that require special configurations.

REFERENCES

- Resources:
  - https://dash.plotly.com/
- Figures:
  - Fig 1: Screenshot of quantitative feedback summary for advisors.
  - Fig 2: Logo images of tools used.
  - Fig 3: Screenshot of advisor dashboard.
  - Fig 4: Screenshot of comparative chart on advisor dashboard.
- Mentor: Dr. Jonathan Duncan
- Gitlab Repository: https://gitlab.cs.wallawalla.edu/wwu/advisor-survey