ONLINE INTERNSHIP PLATFORM

Presentation by Nilhan Topcu and Sule Ilkbakar

INTRODUCTION

The **Online Internship Platform** is an online platform aimed at enhancing students' internship experiences. The primary objectives of the application are to improve efficiency and transparency in the internship process.

- To enable the evaluation of internships.
- To provide a detailed overview of each student's internship progress.
- To establish communication channels between students, academic institutions, and companies.
- To enable students to upload and manage internship documents directly through the platform's document management system.

COMPLEX INTERNSHIP PROCESS

- Students obtained internship forms from the university website and sent them via email.
- This process caused challenges for students, departments, faculty deans, and internship coordinators.
- **Solution:** A process map with step-by-step explanations was created on the homepage.

COMPLEXITY OF DOCUMENTATİON

- Students had difficulty finding and manually completing the necessary documents, which made the process complex.
- **Solution:** A document management system was implemented to make documents easily accessible and automatically fill in required sections.

COMMUNICATION CHALLENGES

- Constant communication between authorities and students was required during the internship process.
- This could lead to misunderstandings and delays.
- **Solution:** Students can now monitor the status of their internship processes through the system.

NestJS Framework

- NestJS is an efficient framework built on Node.js.
- Written in TypeScript.
- Its modular structure, high performance, and robust testing support make it easier to write code.

Fetch Method

- The Fetch method is a modern JavaScript API for retrieving data and handling HTTP requests and responses asynchronously.
- It improves application performance by retrieving data in the background without locking the user interface.

Bootstrap Library

- Bootstrap is a popular CSS library that accelerates the web development process.
- It offers user-friendly design components.

Database Management and PostgreSQL

- PostgreSQL is an open-source database management system that offers reliability, performance, and extensibility.

TypeORM

- TypeORM is a Node.js ORM module that offers strong type support and flexible configuration.
- It supports various databases and uses decorators to define data models and relationships.

LITERATURE REVIEW

Features/Platforms	Web-Based Internship Management System (UiTM Perlis)	Industrial Training System (UTM)	Industrial Training System (UNIMAS)	FTK Industrial Training System (UTeM)	Internship Application System (IAS)	Internship Monitoring and Supervising System (iMAPS)	Web Development (Internship)
User Categories	Student, Coordinator, Employer	System Administrator, Committee, Faculty Member, Student	Student, Coordinator, Employer	Manager, Student, Industry	Student, Coordinator	Student, Coordinator, Industrial Supervisor, Visiting Professor	App owners
Recommendation System	Yes	No	No	No	Yes	No	Yes
Automatic Placement	Yes	Yes	Yes	Yes	Yes	No	Yes
Usability and Interface Evaluation	Very Good	Good	Good	Very Good	Very Good	Very Good	Very Good
Testing and Evaluation	Usability, Expert	-	-	-	Usability	Usability	Agile Methods

USER ROLES AND FUNCTIONALITIES:

STUDENT

- **Internship Application:**Fills out internship information and applies, downloads necessary forms.
- **Uploading Documents:** Uploads completed forms and copies of ID.
- **Tracking Applications:** Monitors the status of internship applications and downloads approved insurance documents.
- **Uploading Reports:** Uploads internship reports for academic review.

DEPARTMENT

- **Reviewing Applications:** Approves or rejects student internship applications.
- **Evaluating Reports:** Reviews and evaluates uploaded internship reports.
- **Viewing and Evaluating Companies:** Evaluates companies where students have completed their internships.

FACULTY DEAN

- **Approving Internships:** Reviews and makes final decisions on internships approved by the department.
- **Viewing Companies:** Views companies evaluated by the department.

INTERNSHIP COORDINATOR

- **Tracking Applications:** Monitors the status of student internship applications.
- **Uploading Insurance:** Manages and uploads insurance documents.
- **Filtering Internships:** Filters internships by status and manages internship processes.
- **Viewing Companies:** Views companies evaluated by the department.

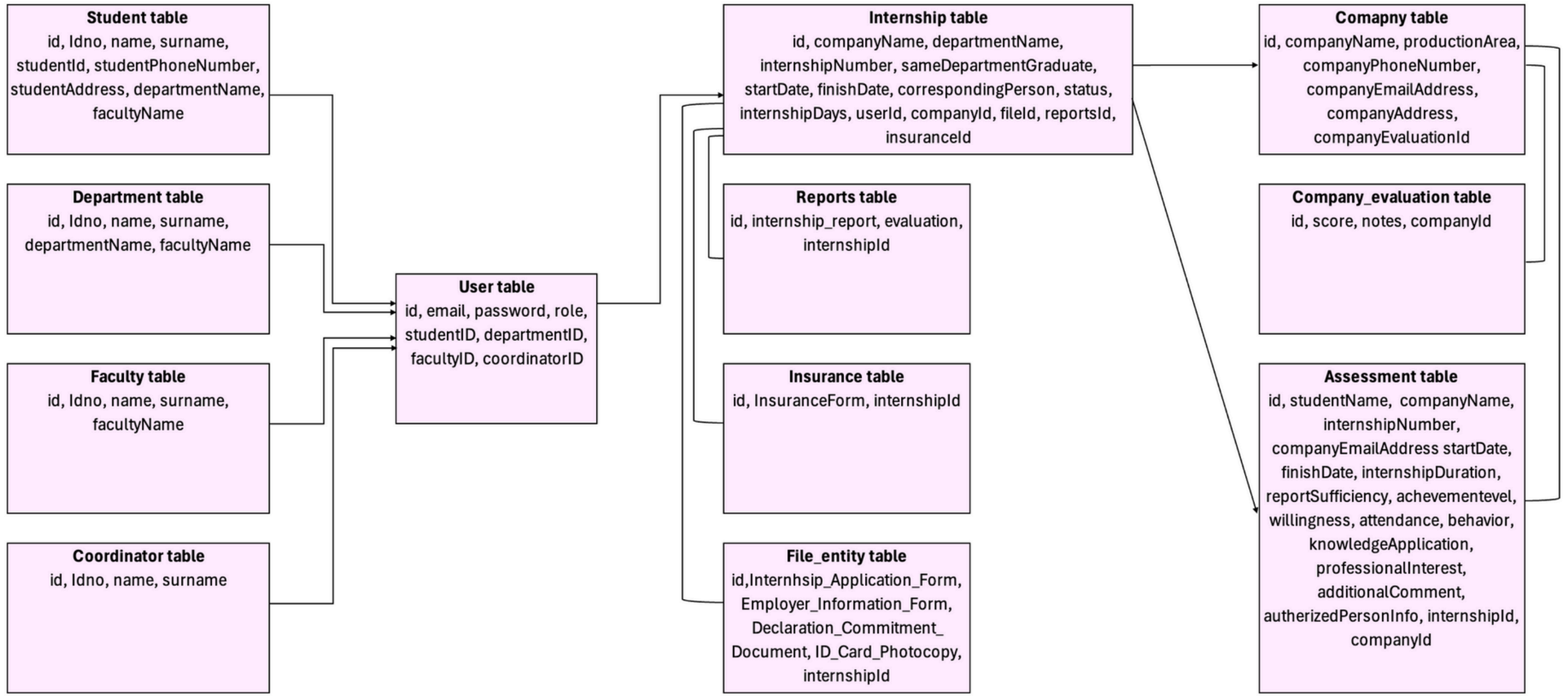
SYSTEM ARCHITECTURE

High-Level Architecture

The "Online Internship Platform" follows the Model-View-Controller (MVC) architecture, which separates the application into three main components: Model, View, and Controller. This structure helps in organizing the codebase and makes it easier to maintain and scale.

- **Model:** Manages data processing and database operations. In our project, we used PostgreSQL and TypeORM for this purpose.
- **View:** Displays the user interface and presents data to the user. In our project, we used HTML, CSS, and JavaScript for this purpose.
- **Controller:** Mediates between the Model and the View, and processes user inputs. In our project, we used NestJS for backend operations.

SYSTEM ARCHITECTURE



BACKEND DEVELOPMENT

A new NestJS project was created.
Authentication was established, and internship modules were designed.
PostgreSQL and TypeORM were configured for database management.
Entities were defined for database tables and relationships were established.
CRUD operations were implemented through controllers and services.
Modules were organized in the app.module.ts file to ensure a scalable and modular design.

FRONTEND DEVELOPMENT

After the backend was organized, the design process was initiated.
Login pages for students and users were developed.
The Fetch method was used to establish a connection between the frontend and backend.
A user-friendly and responsive interface design was provided in accordance with modern web standards.

TECHNOLOGIES USED

BACKEND

- NestJS: For scalable and maintainable backend structure.
- TypeORM: For managing database operations through TypeScript.
- PostgreSQL: For handling complex database operations.
- Passport and JWT: For user authentication and session management

FRONTEND

- HTML, CSS, JavaScript: For creating a responsive user interface.
- Fetch API: For HTTP requests to backend.

DOCUMENT MANAGEMENT

- File upload and download functionalities.
- Document validation before upload.

OTHER MODULES:

- Multer: For secure file uploads.
- Schedule: For managing cron jobs and scheduled tasks.

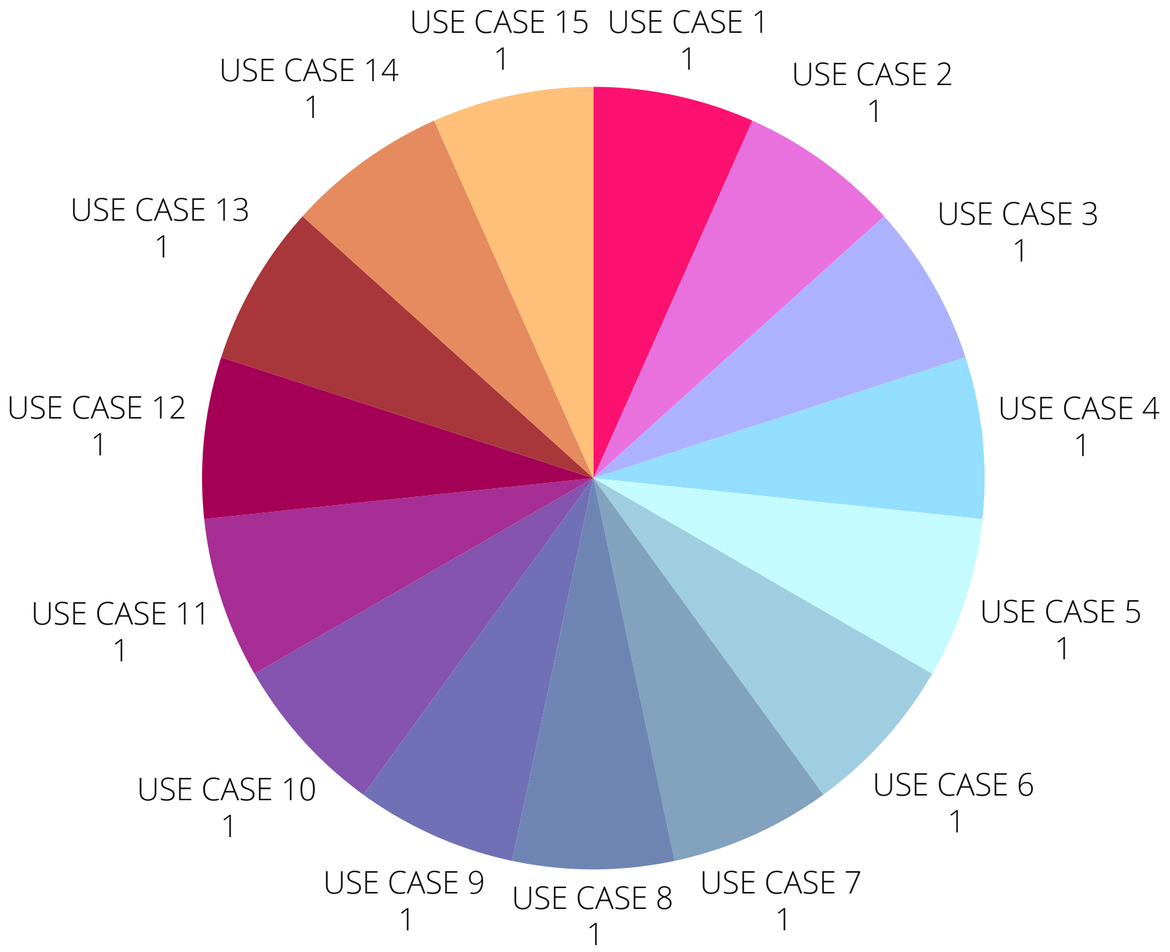
RESULT

USE CASES	Actors	Steps	Result
Approval of Internship Application by the Faculty Dean	Faculty Dean	Manager logs in, clicks 'Internship Processes,', reviews and approves application.	SUCCESSFUL/ PASSED
Rejection of Internship Application by the Faculty Dean	Faculty Dean	Manager logs in, clicks 'Internship Processes,' , reviews and rejects application.	SUCCESSFUL/ PASSED
Review of Companies	Faculty Dean	Manager logs in, clicks 'Companies, ' reviews companies.	SUCCESSFUL/ PASSED
Entering Insurance Information and Uploading Documents	Internship Coordinator	Coordinator logs in, clicks "Internship Processes, " uploads insurance document.	SUCCESSFUL/ PASSED
Filling Out the Internship Evaluation Form	Internship Supervisor	Supervisor clicks evaluation form link, fills out and submits form.	SUCCESSFUL/ PASSED
Review of Internship Evaluation Form	Academic Unit Personnel	Staff logs in, clicks "Internship Processes,", reviews evaluation form.	SUCCESSFUL/ PASSED
Viewing Companies Where Internships Were Completed	Internship Coordinator	Coordinator logs in, clicks "Companies, " views companies.	SUCCESSFUL/ PASSED

USE CASES	Actors	Steps	Result
Appplication for Internship approval	Corresponding Person, Student	Student login, selects "Internship Application", enters instution info, fills out and submit	SUCCESSFUL/ PASSED
Tracking Internship Application Processes	Student	Student logs in, selects "My Internship Applications," , views status and approval processes.	SUCCESSFUL/ PASSED
Uploading Internship Reports	Student	Student logs in, selects "Internship Reports," uploads report in PDF format.	SUCCESSFUL/ PASSED
Approval of Internship Application by the Academic Unit	Academic Unit (Department)	Staff logs in, clicks "Internship Processes," , reviews and approves application.	SUCCESSFUL/ PASSED
Rejection of Internship Application by the Academic Unit	Academic Unit (Department)	Staff logs in, clicks "Internship Processes,", reviews and rejects application.	SUCCESSFUL/ PASSED
Review of Internship Reports	Academic Unit (Department)	Staff logs in, clicks "Internship Processes,", downloads and reviews reports.	SUCCESSFUL/ PASSED
Evaluation of Internship Reports	Academic Unit (Department)	Staff logs in, clicks "Internship Processes," , evaluates and grades reports.	SUCCESSFUL/ PASSED
Review and Evaluation of Internship Companies	Academic Unit (Department)	Staff logs in, clicks "Companies,", reviews and evaluates companies.	SUCCESSFUL/ PASSED

Quantitative results such as accuracy, speed, or efficiency can be directly observed in the terminal during the code demonstration. Since including all quantitative data in the slides would take up considerable space, it has been reserved for the code presentation session.

RESULT



- We tested our use cases and assigned a value of 1 if the result was passed, and a value of 0 if the result was failed.
- As can be seen, all our use cases were successfully passed.

CHALLENGES

INTERNSHİP STATUS MANAGEMENT

- We faced difficulties in planning and implementing the statuses of internships. When we completed one page, we encountered status errors on another page.

SOME DATABASE TABLE CREATION AND METHODS:

- Creating the insurance, report, and file_entity tables in the database and adding download and upload methods were challenging. We frequently received unauthorized errors when attempting to download files.

AUTOMATED EMAİL ON INTERNSHİP COMPLETİON

- We had difficulties in sending an automatic email to the company at a specific time when the internship status changed to 'completed'. Writing the method for this process was challenging, and we conducted online research to solve it.

FUTURE WORK AND CONCLUSİON

Most of the design goals for this project have been accomplished. Below are the primary obstacles faced and how they were resolved:

- Efficiency: The system's design enhances user interaction and productivity.
- Security: Precautions were taken to protect user data, with role-based access control implemented.
- Documentation and Auto-Completion: Functions for document management, assessment, and auto-completion were successfully implemented.
- Internship Tracking: Assigned statuses to internships based on approval or denial by authorities.

Future work:

- Mobile app support
- Advanced Analytics
- Internationalization
- AI and MachineLearning
- Security Enhancemets

REFERENCES

REFERENCES 1

Abdullah, FazeeraSyuhada, et al. "Web-Based Internship Management System." Journal of Information Systems, vol. 15, no. 2, 2022, pp. 123-134.

REFERENCES 2

Ibrahim, Alif Faisal, et al. "Internship Application System Using Laravel PHP Framework." International Journal of Web Applications, vol. 18, no. 1, 2021, pp. 45-57.

REFERENCES 3

Amron, Mohd Talmizie bin, et al. "Development of Internship Monitoring and Supervising Web-Based System for Electrical Engineering Students." Journal of Educational Technology & Society, vol. 25, no. 3, 2023, pp. 89-101.

REFERENCES

REFERENCES 4

Pham, Anh Duc. "Web Application Development Using NestJS." Bachelor Thesis, University of Technology, 2023.

REFERENCES 5

Santo, Guilherme Ferreira. "Web Development: Internship Report." Master's Degree in Mobile Computing – Computer Science, Polytechnic Institute of Leiria, 2023.

REFERENCES 6

NestJS. (n.d.). A progressive Node.js framework. Retrieved from https://nestjs.com/

REFERENCES 7

Mozilla Developer Network. (n.d.). Fetch API. Retrieved from https://developer.mozilla.org/en-US/docs/Web/API/Fetch_API

REFERENCES

REFERENCES 8

Bootstrap. (n.d.). The most popular HTML, CSS, and JS library in the world. Retrieved from https://getbootstrap.com/

REFERENCES 9

PostgreSQL. (n.d.). The World's Most Advanced Open Source Relational Database. Retrieved from https://www.postgresql.org/

REFERENCES 10

TypeORM. (n.d.). An ORM for TypeScript and JavaScript. Retrieved from https://typeorm.io/

THANK YOU FOR LISTENING