Nazarbayev University

Computer Science Department
Final Report – Spring 2025

Title of the project:	"Communication Channel"
Team Members:	Aruzhan Sarsenbayeva, Nursat Zeinolla, Arsen Aisin, Ilyas Temirbayev, Kainar Nurlanbek
Project Advisor/Co-Advisors	Askar Boranbayev, Kuanysh Yersakhanov

Executive Summary (10%)

This project sought to create a university communication platform designed to alleviate fundamental problems students encounter while trying to reach faculty or administrative departments like vague procedures, elongated response times, and concealment of information.

The primary aims pinpointed included:

- Establish a complete system for student queries and responses.

- Facilitate the automatic classification of tickets and their assigned departments.

- Allow tracking the progress of requests.

- Provide an interface that is simple and easy to navigate.

Methodology

We used a systematic approach to software development beginning with requirements analysis followed by wireframing and UI/UX designing, then implementing the frontend in React, backend using Java Spring, and database in PostgreSQL. We adopted incremental design, with continuous testing and refinement based on evolving feedback.

The evaluation was completed through a closed trial with students only. A total of 15 own participants interacted with the system in addition to completing a survey to evaluate its usability and effectiveness.

Main Outcomes

- Easy to use, solving communication problems rated at 4.4/5 and 4.1/5 respectively.

- The resolution of centralization in communication, eased responsiveness, and satisfaction.

- Enhanced communication systems provided by the feedback guaranteed an improvement to the most

existing communication systems used.

Alignment with Computing-Based Solution

This project demonstrates clearly the computing-based solution's design, implementation, and evaluation. It required problem solving by software engineering, database development, and user interface design. The project provided a working prototype and verified its usefulness with a user evaluation, achieving important educational and professional objectives.

Introduction (10%)

Problem statement:

We suggest a unified communication platform that helps to erase fragmented, inefficient interactions between students, faculty and administrative staff. In many cases, feedback, complaints, and inquiries are directed to different channels and are delayed, miscommunicated, and have no one to blame. Centralized communication system must exist, which may act as a 'common room' to have all interactions in a streamline manner, categorize and distribute accordingly while keeping them confidential.

Envisioned Solution:

Therefore, we suggest developing a holistic software platform that will act as the university's only communication channel. This will be a centralized platform to collect all feedbacks and complaints, internal communications, and queries of the university by making this easier, faster, smoother, and safer than ever. It is set to be installed locally within the university's IT infrastructure, thus eliminating the concern of data breach and complete confidentiality.

Background and Related Work (15%)

Efficient communication systems are integral to large organizations, including educational institutions. Current research highlights the benefits of centralized platforms in reducing fragmentation and improving organizational efficiency. Examples from industry include ticketing systems like Zendesk and open-source solutions like GLPI. While these solutions have inspired this project, the unique requirements of a university setting, such as academic, administrative, and student-specific communication needs, necessitate a tailored approach.

Moreover, related projects within educational contexts reveal the value of integrating intelligent ticketing systems and knowledge bases to empower users and reduce dependency on manual intervention. Lessons from these implementations will inform the proposed solution, ensuring adherence to best practices while addressing the university's regulatory and operational requirements.

Project Approach (20%)

A structured and collaborative approach by the system design, technology selection, modular development and comprehensive testing of the University Single Communication Channel Software development has occurred. This section describes the technical architecture, the key components used in the project, the methodologies for development and the strategies the team adopted in collaboration.

1. Solution Architecture

The system was designed with a modular layered architecture in order to maintain scaleability, the maintainability, and the security of the system. It includes:

Frontend: React.js for responsive, dynamic interfaces.

Backend: Java Spring for robust and scalable server-side logic.

Database: PostgreSQL with Pgsodium for secure data storage.

Security: Role-based access, SSL encryption, and compliance with regulations in Kazakhstan.

Hardware Environment: It is intended to be locally installed on the university's internal IT infrastructure. The servers must support:

Horizontal and vertical scaling

High concurrency (minimum 1,000 simultaneous users)

Encrypted HTTPS connections via SSL

Role-based access control

2. System Workflows and Features

Workflows:

User Feedback, Complaints or Inquiries: Users provide complaints, feedback or inquiries and they will be auto classified and stored in the database.

Assignment: The assignment of tickets to staff based on category, expertise under the criteria of workload.

Direct Messaging: Mails can be addressed to specific employees or departments.

Assigned staff are accountable to reply to and resolve tickets, analytics are produced for keeping track of.

Features:

Multilingual interface (Kazakh, Russian, English)

Knowledge Base integration for self-service

Analytics dashboard for admins

Secure document handling and confidentiality enforcement

Displays support for role based user types (students, faculty, admins)

3. Third-Party Components and Integration

Several third-party libraries and tools were utilized to accelerate development and ensure quality:

-Spring Boot/Laravel: Used as the backend framework to provide routing, middleware, and service handling.

-React: A popular JavaScript library for building responsive UIs.

-PostgreSQL: Robust open-source database for structured data storage.

-Localization Libraries: Tools for supporting dynamic translation and multilingual data entry.

-Security Tools: SSL libraries, RBAC modules, and hashing utilities for password and data protection.

-Analytics and Charting: Third-party libraries (e.g., Chart.js or Recharts) were used to visualize data on dashboards.

Integration involved standard API-based communication between modules. All external components were thoroughly evaluated for compatibility, security, and licensing before integration.

4. Team Collaboration and Workflow

The project team of five students functioned effectively by adopting agile methodologies and assigning clear roles:

a. System and Design tasks: (Aruzhan, Kainar, Arsen)
1. Finalize system architecture, including database schema and service endpoints
2. Fix some UI/UX corrections

b. Functionality tasks: (Nursat, Ilyas)
1. Implement user authentication and authorization (JWT-based)
2. Develop student ticket submission module
3. Implement staff ticket handling features
4. Develop admin user management functionalities

c. Intelligent ticketing: (Arsen, Nursat, Ilyas)
1. Develop intelligent ticket distribution using past satisfaction data and staff workload
2. Create a push notification system for ticket updates

d. Statistics and Analytics tasks: (Aruzhan, Kainar)
1. Build analytics page with real-time metrics
2. Show users/tickets based data in user-friendly design

e. System Integration & Enhancements tasks: (Aruzhan, Kainar, Arsen, Nursat, Ilyas)
1. Implement Nazarbayev University Gmail OAuth authentication.
2. Develop Knowledge Base (FAQ section for self-service support)
3. Improve multilingual support (Kazakh, Russian, English)

f. Security & Compliance: (Nursat, Arsen, Ilyas)
1. Ensure data encryption (Pgsodium) for sensitive information.
2. Implement role-based access control.
3. Align with Kazakhstan’s data protection laws.

Project Execution (15%)

In the past two semesters, our team worked on creating a communication software for a university aimed to enhance the communication among students, staff, faculty, and administrative personnel. The project commenced with a research and planning stage, where we tried to understand the communication difficulties in educational institutions and documented fundamental issues. Initially, we agreed to implement a feedback form, smart ticket routing system, secure messaging system, an analytics dashboard, and a knowledge base.

Our initial thoughts were to build the platform's backend on Java Spring and the frontend using JavaScript. In the early development stages, we realized that React.js was more scalable and had better component reusability, leading us to switch to a React + Java Spring stack. This shift led to faster and smoother development on the frontend as well as overall platform redundancy.

Some features also evolved based on feedback from our supervisor and peers. For example, we initially planned a basic contact form which later evolved into a ticket system with status updates and automatic routing based on keywords to appropriate departments. The system is now much more intelligent and user-friendly despite the added complexity.

What Went Right/Wrong

In this case, what went right:

- Group division and the planning stage outline proved useful.

- Weekly check-in meetings allowed effective monitoring of planned milestones.

- Development of UI and development testing was quicker with the adopting of React.

What went wrong:

- Underestimating the complexity of the backend ticket routing logic.

- Postgres query challenges during the implementation of the analytics.

- Communication drop off during semester breaks caused some milestone delays.

Stricter deadlines—dividing complex tasks into parts—helped overcome challenges. Backend issues were resolved with the input of our supervisor around queries and data structuring. The group was also able to make up for the delays through Trello outfitted task management.

Teamwork & Roles

Collaborative strategies employed included pair programming, group testing sessions, and weekly retrospectives. Solutions were sought collectively, and discussion was largely done by voting after everyone had a say.

Generally, the execution phase was both difficult and informative. We acquired invaluable insights in practical project management, technical execution, as well as teamwork within time constraints.

Evaluation (20%)

In order to evaluate if our university communication platform resolved the problems around information delays, feedback channels, and separation of information, we conducted a focused evaluation with only student participants.

Evaluation Methodology

We invited a total of 15 students from different faculties in the university and asked them to help us test the system. Each student was tasked with several important activities:

- Submit feedback to a chosen department.
- Monitor interaction with ticket responses.
- Look up relevant information from the knowledge base.
- Provide an overall rating of the platform.

All these participants were able to complete these tasks successfully and they all submitted responses using the System Usability Scale (SUS) as well as custom questions regarding usability, effectiveness, impact, and overall system evaluation.

Information Obtained

The following includes the summary of the responses (average Score (out of 5)):

- Ease of submitting a request - 4.4
- Clarity of the user interface - 4.6
- Usefulness of ticket tracking - 4.2
- Effectiveness in solving issues - 4.1
- Overall satisfaction - 4.3

Important Insights

Students enjoyed the clean user interface along with the ease of navigation.

From a sample size of fifteen students, thirteen stated that the platform was more effective in comparison to traditional communication through emailing or visiting in-person.

Some students offered suggestions for improvement like automated notifications for status updates.

Validation of the Solution

This focused evaluation from the students within the target group confirms the following regarding the platform:

Enhances the rate and understanding of communication between students and university offices.

Streamlines the process of reporting and tracking issues and gives students the impression that their voices are accounted for.

User experience during this period was positive for students when compared to their previous experiences. With the survey results and provided comments taken into consideration, we can ascertain that the system

successfully resolves the communication concerns of students and supports our automated solution approach.

Conclusion and possible future work (5%)

As discussed above, our group has successfully implemented a communication platform to assist students within a university setting. The system employs a multifaceted approach to solve underlying issues of slow replies, ambiguity, and disintegrated communication channels. The system is a comprehensive solution that offers functions such as ticketing, intelligent routing, and a user-friendly interface to enhance student-department interactions.

Out of the 15 students we evaluated, the overwhelming feedback received showcased that the platform was simple to use, efficient, and generally appreciated. The majority of users agreed that the platform enhanced the speed and clarity of communication relative to other existing methods. This demonstrates that our solution does effectively help in solving computing-based problems within a university environment.

Possible Future Work

Although the core goals aimed at were achieved, it is essential to note the following proposed improvements.

- Mobile Optimization: Developing a mobile application or enhancing system responsiveness for easier smartphone access.
- Live Chat Integration: Permitting faster resolutions for issues that do not require formal ticketing.
- Real-Time Notifications: Issuing email or in-app notifications regarding any changes made to the ticket.
- AI Ticket Categorization: Improvements to smart-routing by incorporating machine learning for increased accurate classification.
- Multilingual Support: Expanding language options to cater to a wider audience.

Such additions would improve usability, accessibility, and operational efficiency. The system has the ability to advance communication within universities as the platform's development and growth continues.

References (5%)

Wagner, D., & Schneier, B. (1996, November). Analysis of the SSL 3.0 protocol. In *The Second USENIX Workshop on Electronic Commerce Proceedings* (Vol. 1, No. 1, pp. 29-40).

Alneyadi, S., Sithirasenan, E., & Muthukkumarasamy, V. (2016). A survey on data leakage prevention systems. *Journal of Network and Computer Applications*, 62, 137-152.

Mantelero, A. (2016). Personal data for decisional purposes in the age of analytics: From an individual to a collective dimension of data protection. *Computer law & security review*, 32(2), 238-255.

Abdullah, H. M., & Zeki, A. M. (2014, December). Frontend and backend web technologies in social networking sites: Facebook as an example. In 2014 *3rd international conference on advanced computer science applications and technologies* (pp. 85-89). IEEE.