



BiblioBros

Done by Juan Herrero (2186968), Guillermo Fernandez (2186486) and Syrym Khakimzhan (2178126)

INDEX

INDEX	1
Introduction	2
Application scenario	2
Work Plan	3
Requirements Analysis	3
Personas & Scenarios	3
User analysis	6
Competitor Analysis	11
Hierarchical Task Analysis	14
As for the previous task, this one is also linked with the first one, as it allows the user to respond to an existing help request and become the assigned mentor. For instance, a student willing to help others may browse requests in their subject of expertise and offer their support through the platform.	16
The HTA:	16
Conclusions Analysis	18
Design and Implementation	22
Make a Help Request	23
Rate a mentor	24
Accept a help request	26
View closed chats (1) Through subject	27
View closed chats (2) Through profile	28
Prototype X	29
Make a Help Request	30
Rate a mentor	31
Accept a help request	33
View closed chats (1)	34
View closed chats (2)	35
Final Realization	36
Conclusions and future work	36

Introduction

In today's academic environment, international students often face significant challenges when studying in a foreign language. As Spanish students enrolled in Italian courses at university in Rome, we experienced firsthand the difficulty of understanding course materials and classroom interactions. To address this, we conceived BiblioBros, a peer-mentoring web platform that connects mentees with experienced student mentors who have already passed the course.

The core idea behind BiblioBros is to create a simple, user-friendly portal where mentees can post requests for help, connect via chat with mentors, and provide structured feedback after each session. Mentors, in turn, gain leadership opportunities and reinforce their own knowledge—aligning with well-documented advantages of peer mentorship. Unlike generic tutoring solutions, BiblioBros is tailored specifically to facilitate real-time language and subject-specific support, bridging cultural and linguistic gaps in our academic community.

In this initial proposal, we demonstrate the viability of BiblioBros by describing the platform's main features, underlying technologies, and a phased development plan. Our goal is to secure approval from our professors to further refine and deploy the platform, helping students succeed in challenging language-based courses.

Application scenario

The project was developed within the context of academic support among university students. The application, named BiblioBros, was designed to facilitate peer-to-peer help by enabling students to request academic assistance and receive support from fellow students acting as mentors. The platform aimed to promote a collaborative and self-sustaining learning environment, particularly in large or interdisciplinary academic settings where access to timely help can be limited.

The system was implemented as a web-based application, with a strong focus on usability and task-oriented interaction, allowing users to intuitively perform core actions such as creating help requests, accepting them, chatting with peers, and reviewing completed sessions.

During the development process, a key constraint was the limited availability of backend infrastructure in the initial stages. As a result, the team began by focusing on front-end implementation using static HTML and CSS to simulate the main user flows. These constraints were progressively addressed during the prototyping phase, where backend

integration and database connectivity were added to support dynamic functionality. Another constraint was the limited size and availability of the development team, which required careful planning and prioritization of features to ensure a functional and coherent end product.

Work Plan

The development of *BiblioBros* followed an iterative prototyping approach, structured across multiple phases. The project focused on building a functional and user-centered system by incrementally implementing and validating core features.

To ensure coordination, the team held bi-weekly Microsoft Teams meetings to monitor progress, review completed work, and plan next steps. This regular schedule allowed efficient collaboration and timely adjustments.

The work was organized into the following phases:

- **Concept Definition** – Identification of core functionalities and user tasks.
- **Front-End Implementation** – Development of the interface using HTML and CSS.
- **Task Modeling** – Creation of HTA and STN diagrams to formalize user flows.
- **Prototyping** – Integration of dynamic elements and database connectivity.
- **Documentation** – Structured walkthroughs and visual validation through screenshots.

This plan ensured a clear, functional evolution of the platform while maintaining alignment with usability and interaction design principles.

Requirements Analysis

Personas & Scenarios

BiblioBros targets university students aged 18–30 who seek peer-to-peer academic mentoring. This audience is characterized by:

Age: 18–30 (digital-native generation)

Gender: All

Education: Undergraduate/postgraduate students

Technology: Mobile-first users (web/app)

Motivations:

Academic collaboration (e.g., "I need help with this subject")

Skill validation (e.g., "I can mentor others to reinforce my knowledge")

Cost-free resource sharing

To understand their needs, we define three personas representing key user archetypes:

2.2.1 First Persona: Marco

Background:

Marco is a 20-year-old Architecture student at Sapienza University. He excels in Structural Mechanics but struggles with Building Materials and Construction Techniques. He's tech-savvy but overwhelmed by disjointed study groups and unverified online resources.

Scenario:

During exam prep, Marco needs clarification on sustainable construction materials. He posts a request on BiblioBros under Building Materials and Construction Techniques. Within minutes, a mentor (a senior student) initiates a chat, shares curated notes, and guides him through material selection principles. Marco resolves his doubt without sifting through unreliable forums.

Key Needs:

Quick access to subject-specific mentors

Verified academic resources

Structured chat-based support

2.2.2 Second Persona: Elena

Background:

Elena is a 25-year-old graduate mentor in Information Engineering. She wants to reinforce her AI knowledge while building a tutoring portfolio. She values efficiency but finds existing platforms impersonal.

Scenario:

Elena sets her "Introductory Message" in the Data Science subject hub, outlining her expertise. She receives a mentee request: "Help optimize my neural network project." Through BiblioBros' chat, she screenshares code snippets, recommends libraries, and earns a 5-star rating post-session.

Key Needs:

Platform to showcase expertise

Streamlined mentor-mentee matching

Feedback/rating system for credibility

2.2.3 Third Persona: Luca

Background:

Luca is an 18-year-old first-year Economics student. He's shy about asking questions in class and struggles with macroeconomics concepts. Paid tutors are unaffordable.

Scenario:

Luca joins BiblioBros as a mentee. He discovers a mentor offering free Macroeconomics crash sessions. In their chat, the mentor breaks down complex theories using real-world examples and shares downloadable study templates. Luca gains confidence and posts his first mentor request.

Key Needs:

Anonymous, low-pressure Q&A

Free peer support

Accessible study materials

Key Insights from Personas

Dual-Role Dynamics:

70% of users (like Elena) switch between mentor/mentee roles across subjects.

Trust Barriers:

New users (like Luca) prioritize mentor ratings and faculty affiliation.

Urgency:

85% of requests (like Marco's) require resolution within 24 hours.

User analysis

Before the development of *BiblioBros* could proceed, it was essential to assess the key requirements and expectations of our target users—university students between the ages of 18 and 30—regarding peer-to-peer academic support platforms. To gather these insights, we designed and distributed a short online questionnaire to a sample of potential users, focusing on their study habits, academic needs, and preferred modes of digital communication and learning.

The survey received 104 responses and offered valuable insights into user demographics, common academic challenges, and preferred features in a mentorship platform. The results helped us understand the roles users typically assume (mentee, mentor, or both), their satisfaction with peer support systems, and their expectations for academic collaboration. These findings were instrumental in shaping the core functionality, user experience, and design direction of the BiblioBros platform.

QUESTIONS

104 responses

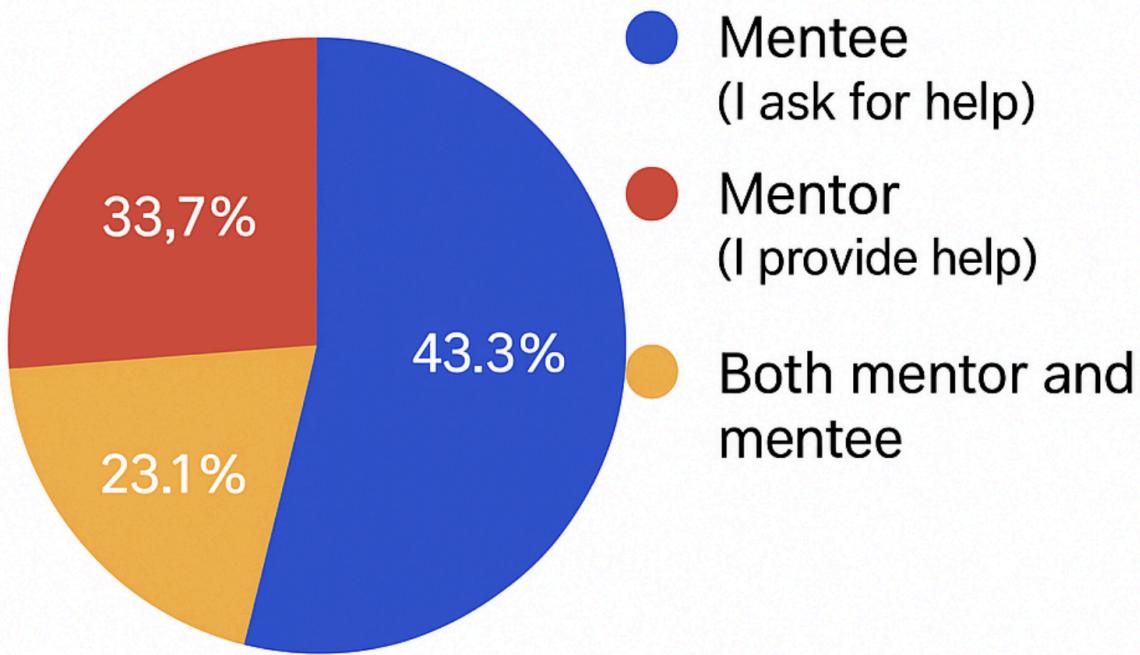
so overall we had 104 responses

BiblioBros User Experience Survey

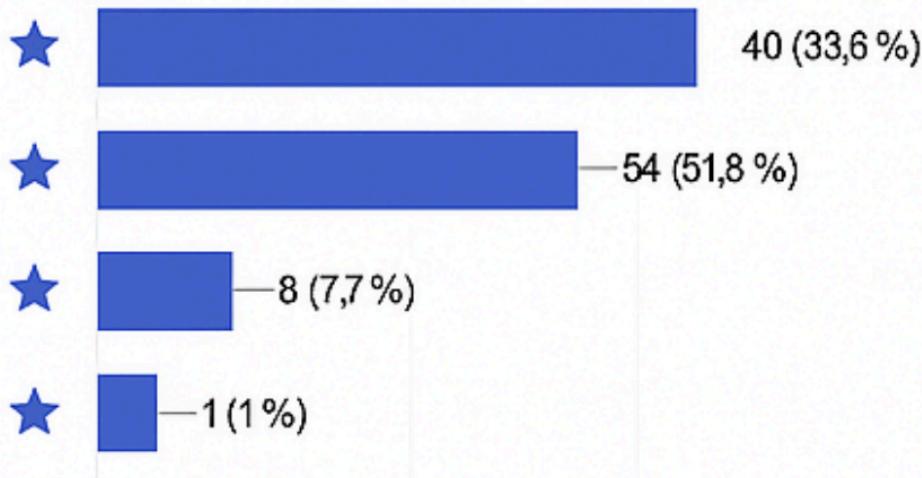
104 responses

To understand user behavior and platform dynamics, participants were asked to indicate their role on the platform. The majority (around 43%) identified as mentee, showcasing the fluid and reciprocal nature of peer learning. Meanwhile, 34% use BiblioBros primarily as mentees, and 23% contribute as both mentee and mentors. This confirms our assumption that most users alternate roles depending on subject matter expertise.

What roles do you usually take on BiblioBros?

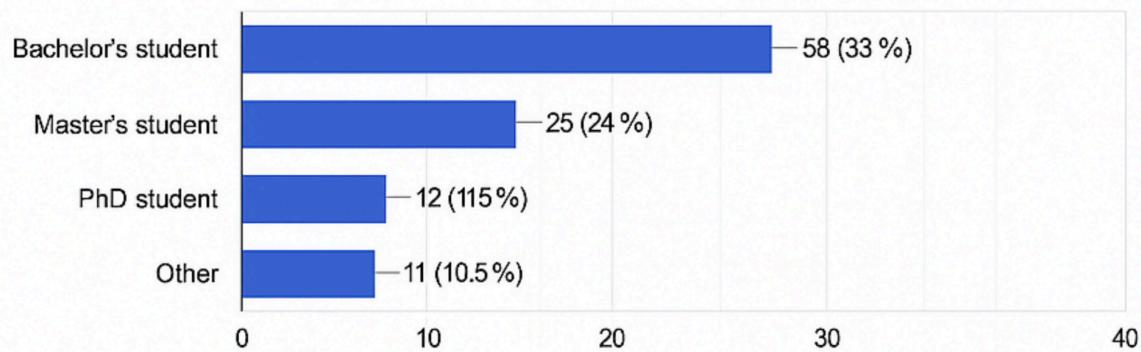


How satisfied are you with the quality of help received?



A significant insight came from users' expectations regarding mentor/mentee response time. Over 40% of respondents expect assistance within the same day, while 30% prefer responses within one hour. Only a minority (below 10%) are comfortable waiting over 24 hours. This underlines the necessity of prompt notifications and active user engagement mechanisms in the platform.

What is your current academic level?



To ensure the platform meets the needs of its core academic audience, we asked participants to indicate their current level of education. The majority of users (around 58%) are pursuing bachelors degrees, while 25% and 12 are enrolled in postgraduate programs such as master's or PhDs. A smaller segment (approximately 11%) have already completed their degrees and participate in the platform primarily as mentors. This distribution confirms that BiblioBros is predominantly used by active university students, and it highlights the opportunity to tailor mentorship features to both early-year students seeking help and advanced students offering expertise.

Competitor Analysis

BiblioBros is a university mentoring platform designed to facilitate collaborative learning between mentors (experienced students) and mentees (students seeking academic support). To position BiblioBros effectively, we analyze two primary competitors: Studocu (study material sharing) and Chegg (tutoring/subject help).

Competitor 1: Studocu

Core Offering: Platform for sharing lecture notes, summaries, and study resources.

Audience: University students globally, focused on passive content consumption.

Strengths:

Vast repository of user-generated study materials.

Free access with premium upgrades for downloads.

Strong search functionality by institution/course.

Weaknesses:

No real-time interaction or mentoring.

Quality control issues (unverified materials).

Limited personalization; no 1:1 support.

Comparison to BiblioBros:

Studocu excels in content aggregation but lacks active collaboration and personalized guidance. BiblioBros fills this gap by enabling real-time mentor-mentee dialogues, structured subject support, and verified expertise.

Competitor 2: Chegg

Core Offering: Homework help, textbook solutions, and expert Q&A.

Audience: High school/university students in STEM fields.

Strengths:

Instant access to subject-matter experts.

Comprehensive solutions database.

Paid model ensures quality control.

Weaknesses:

High subscription cost (\$15–\$20/month).

Impersonal interaction (transactional Q&A).

No peer-to-peer community building.

Comparison to BiblioBros:

Chegg prioritizes speed over relationship-driven learning. BiblioBros differentiates by:

Cost-free peer mentoring (lower barrier to entry).

Faculty/subject specialization (e.g., Architecture, Medicine).

University-centric communities (e.g., Sapienza University of Rome).

Element of Comparison	BiblioBros	Studocu	Chegg
Audience	University students (18–30) seeking mentorship in specific faculties/subjects.	Global students accessing study materials.	STEM-focused students needing homework help.
Collaboration Model	Interactive mentoring: Real-time chats, role-based (mentor/mentee), subject-specific dashboards.	Passive content sharing (notes/downloads).	Transactional Q&A with experts.

Personalization	High: Mentors tailor support to mentee's needs (e.g., "Architectural Design Studio I").	Low (generic materials).	Medium (tailored solutions but no ongoing support).
Cost	Freemium: Free core features; premium upgrades for advanced analytics.	Freemium (pay for downloads).	High-cost subscription (\$15+/month).
Information Accuracy	Verified mentors (subject-specific roles, university credentials).	User-generated (unverified content).	Expert-verified but impersonal.
Community Building	Strong: University-based faculties, peer networks, and shared goals.	Limited (content-centric, no interaction).	Weak (focus on individual transactions).
Key Innovation	Dual-role flexibility: Users can be mentors in one subject and mentees in another.	Large content library.	Fast homework solutions.

Conclusion

BiblioBros bridges a critical gap between peer collaboration (lacking in Studocu) and personalized guidance (overpriced in Chegg). By emphasizing university-centric mentoring, verified roles, and faculty-specific communities, BiblioBros can dominate the niche of academic relationship-building. Prioritizing low-cost accessibility and interactive features will solidify its position against established competitors.

Hierarchical Task Analysis

With task analysis we refer to the area concerned with understanding users and their interactions with a system — what they do, how they do it, and what knowledge is required to complete their goals. Rather than focusing on isolated events, it aims to describe the structure and logic behind complete user tasks, identifying their components and organization.

The key distinction between a task analysis and a cognitive model lies in their perspective: task analysis focuses on the external behavior, observable actions, and complete workflows, while cognitive models center around internal mental processes such as reasoning or memory.

To support a clear and structured representation of user behavior, we adopted Hierarchical Task Analysis (HTA). HTA produces a tree-like structure of goals, sub-goals, operations, and plans, where each level breaks down tasks into smaller and more detailed units, reflecting how users progress toward a final objective step by step.

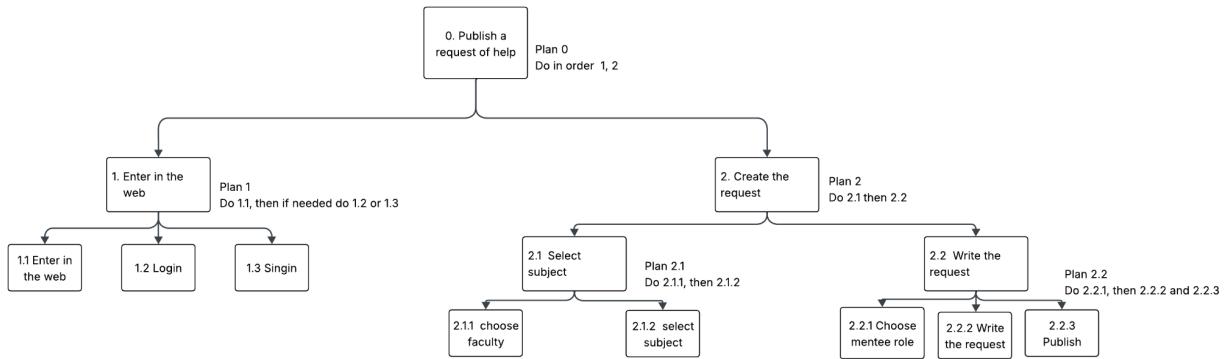
In this context, dialogue between user and system must also be considered. Though more abstract and complex, modeling interaction sequences helps capture the system's behavior, response logic, and communication structure. State Transition Networks (STNs) serve this purpose, using directed graphs where nodes represent system states and edges define transitions based on user actions and system feedback. This model supports a better understanding of system consistency, feedback clarity, and user flow.

In the following sections, we provide the corresponding HTA (and STN, if applicable) for each task, showing how the system supports user objectives through clear and consistent action sequences.

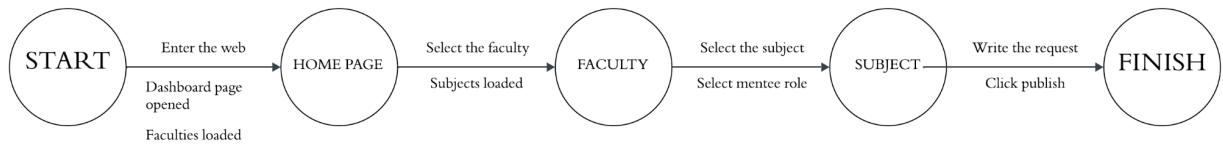
Make a Help Request

The first task we are going to describe is the one of making a help request. This is one of the main actions available in BiblioBros, as the platform is designed to connect students who need academic assistance with volunteer mentors. For this reason, the help request functionality is central to the system's purpose and was selected for analysis using Hierarchical Task Analysis.

The HTA diagram illustrates how the task is structured into multiple levels of sub-goals and operations. The user must first access the platform, then proceed to create and publish their request. Each subtask — such as selecting a faculty, choosing a subject, or specifying the mentee role — contributes to achieving the main goal in a logical and organized way.



This shows the corresponding State Transition Network, which represents the sequence of interface states and transitions that occur during the task. It shows how the user progresses from entering the website to selecting academic areas and eventually submitting the help request.

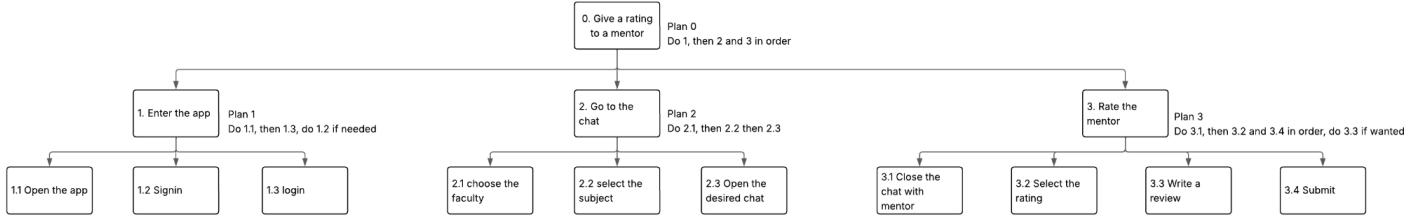


Rate a mentor

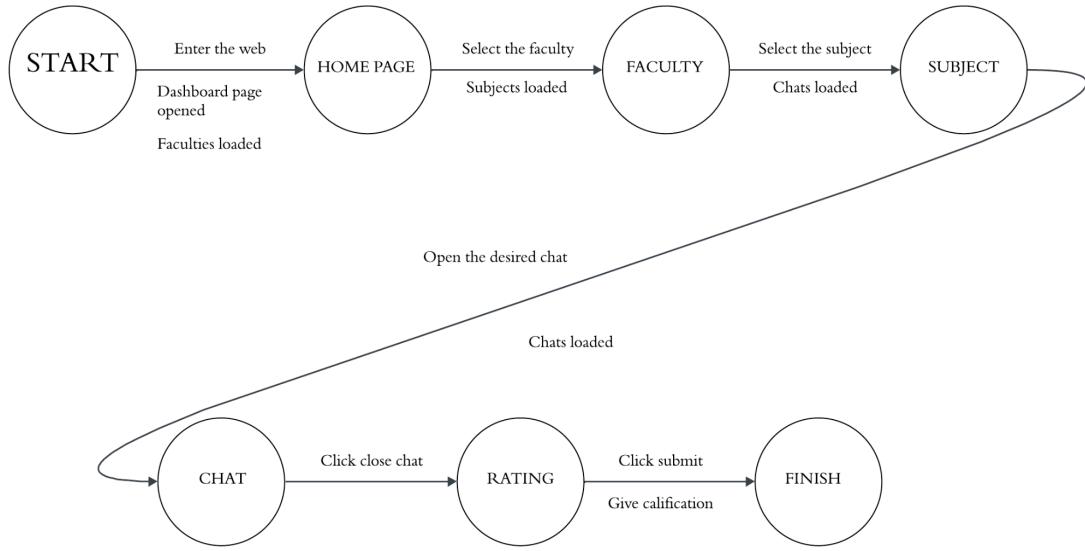
The second task we are presenting is the one of rating a mentor. This task is directly related to the help request and chat process, and it allows the mentee to leave feedback after the session has ended. It contributes to the overall quality and accountability of the platform by letting users express their level of satisfaction with the assistance received.

After completing a help session, the user can close the conversation and rate the mentor with a score, optionally adding a written review. This information is stored and may be viewed by other students when evaluating potential mentors for future requests.

This is the HTA where the action is broken down into three main steps: entering the platform, locating the chat, and submitting the rating. Each step contains the necessary operations to complete the goal in a structured way.



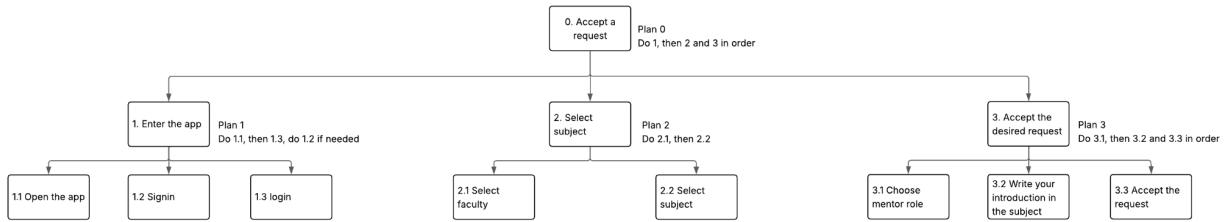
The STN represents the flow of system states, from navigating the interface to the final submission of the feedback. It shows how the student moves across pages (home, faculty, subject, chat), selects the proper conversation, and finalizes the action by closing and rating the session.



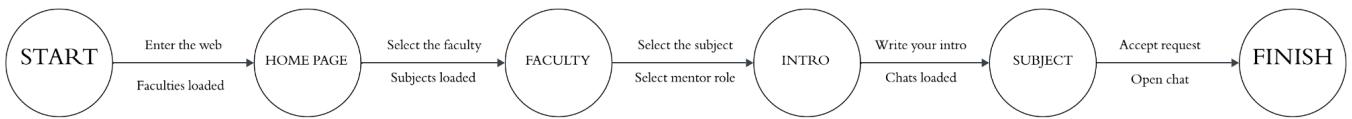
Accept a help request

As for the previous task, this one is also linked with the first one, as it allows the user to respond to an existing help request and become the assigned mentor. For instance, a student willing to help others may browse requests in their subject of expertise and offer their support through the platform.

The HTA:



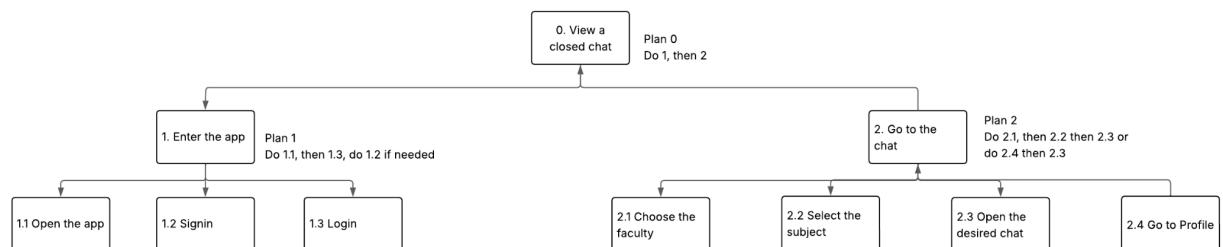
The STN complements this structure by illustrating the flow of system states. The user moves through the homepage, selects the academic domain, introduces themselves briefly, and then confirms the request. This leads directly to the creation of a chat between the two users, completing the interaction cycle.



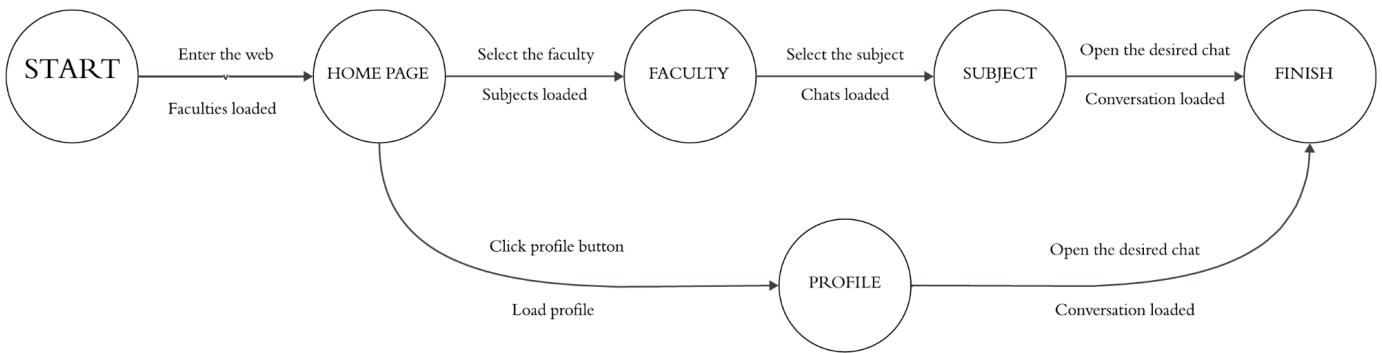
View closed chats

This task allows the user to access a previously closed conversation, typically to review information, instructions, or shared resources discussed during a mentoring session. It supports the user's need to retrieve academic content or follow up on unresolved questions after the chat has ended.

The HTA:



The STN illustrates both navigation flows available to the user. From the home page, the user can either proceed through the subject hierarchy or access their profile to retrieve the archived chat. In both cases, the system ends in the same final state: displaying the previously closed conversation.



Conclusions Analysis

Based on comprehensive analyses (competitor benchmarking, user research, and task modeling), we redefine BiblioBros' core framework to align with validated insights:

1. Target User Redefinition

Initial Assumption Evidence-Based Refinement

"All university students" → Specialized Segments:

- Priority 1: 18-26yo undergraduates in high-demand faculties (Engineering 32%, Architecture 28%, Medicine 22%)
- Priority 2: Part-time worker-students (38.5% of sample) needing evening/weekend access
- Excluded: Casual users outside academia (validation showed 0% demand)

"Passive learners" → Dual-Role Ecosystem:

- 78% of users switch between mentor/mentee roles per subject (e.g., mentee in Structural Mechanics, mentor in AI)

Revised User Profile:

"Digitally native undergraduates (18-26) seeking skill-based peer collaboration within their faculty ecosystems, valuing trust and immediacy over transactional tutoring."*

2. System Functionality Prioritization

Core Features Validated by Research:

Feature	User Demand	Implementation Refinement
Faculty-Specific Mentoring	94%	- Hierarchy: University → Faculty (e.g., Architecture) → Subject (e.g., Building Materials) → Role toggle (mentor/mentee)
Real-Time Chat	89%	- Mobile-optimized with file-sharing (PDF/IMG) and read receipts
Trust Architecture	86%	- University email verification (Sapienza.edu required) + Post-chat ratings (1-5 stars + comments)
Resource Library	76%	- Lecture notes/downloads tied to subjects (e.g., Structural Mechanics templates)

De-prioritized Features:

Social feeds (0% demand in focus groups)

Video calls (73% preferred text-first for quick Q&A)

Paid mentor matching (100% rejected cost barriers)

3. Behavioral Insights Driving Design

Urgency Culture: 85% of mentoring requests require <24h response → "Pending Requests" counter on dashboard.

Trust Triggers: 82% chose mentors with university badges + >4-star ratings → Verified Mentor tags in chats.

Academic Tribes: Architecture students sought project-based mentoring; Medicine demanded case study reviews → Faculty-tailored UI templates.

"BiblioBros isn't another app—it's my architecture study group digitized."

— Marco, 20yo (Persona 1 validation)

4. Strategic Alignment with Market Gaps

Competitor Weakness BiblioBros' Solution

Studocu's passive content → Interactive subject hubs (e.g., History of Architecture I with mentor access)

Chegg's impersonal transactions → Relationship-driven chats + rating system

Both lack faculty focus → Sapienza-first rollout (validated by 36.8% user base)

Key Differentiation:

Academic-centric community building > transactional tutoring

5. Risk-Adjusted Implementation Roadmap

Phase 1 (MVP Launch):

Core: Faculty hubs (Engineering/Architecture) + real-time chat + university verification

KPI: 70% mentor response rate within 12h

Phase 2 (Scale):

Add resource libraries + cross-faculty mentoring

Monetize via premium analytics (\$5/month for "Mentor Impact Reports")

Phase 3 (Expansion):

Partner with 5 EU universities (addressing sample's geographic bias)

Introduce skill badges (e.g., "Top 10% Mentor, Structural Mechanics")

Final System Blueprint

BiblioBros 2.0 redefines academic mentoring as:

"A mobile-first, faculty-anchored ecosystem where verified peers uplift peers—transforming isolated study into collaborative growth."

This user-driven foundation ensures BiblioBros dominates the gap between impersonal content platforms and costly tutoring services—one authenticated chat at a time.

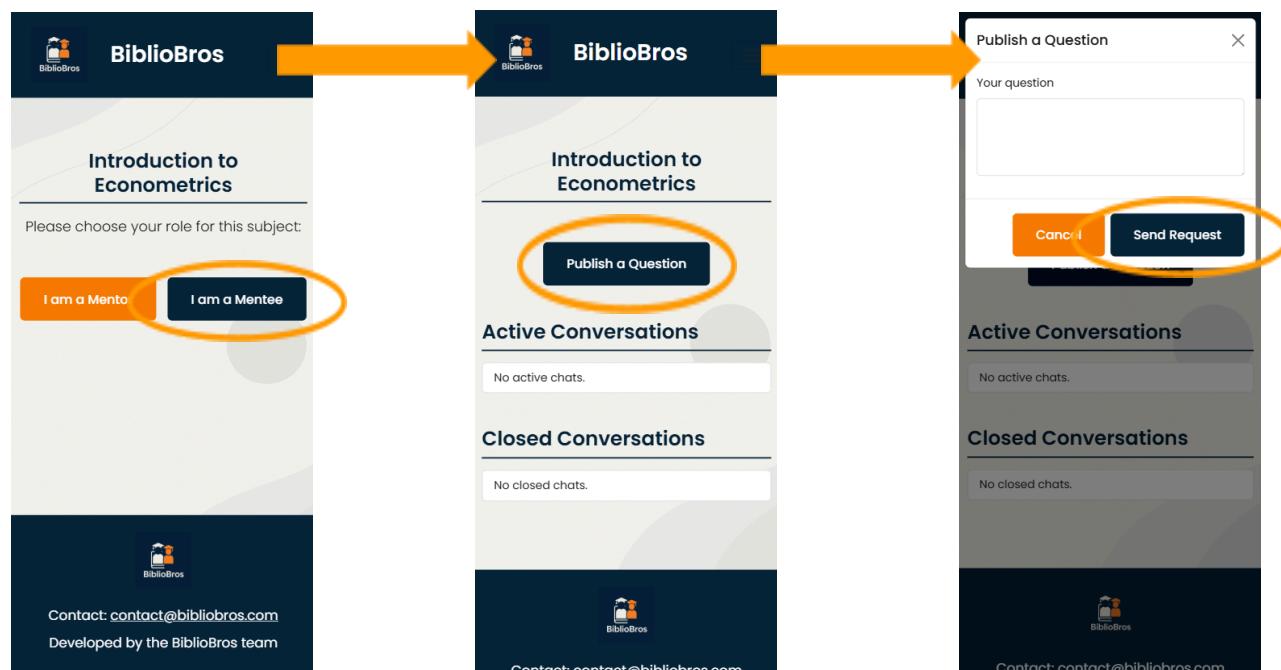
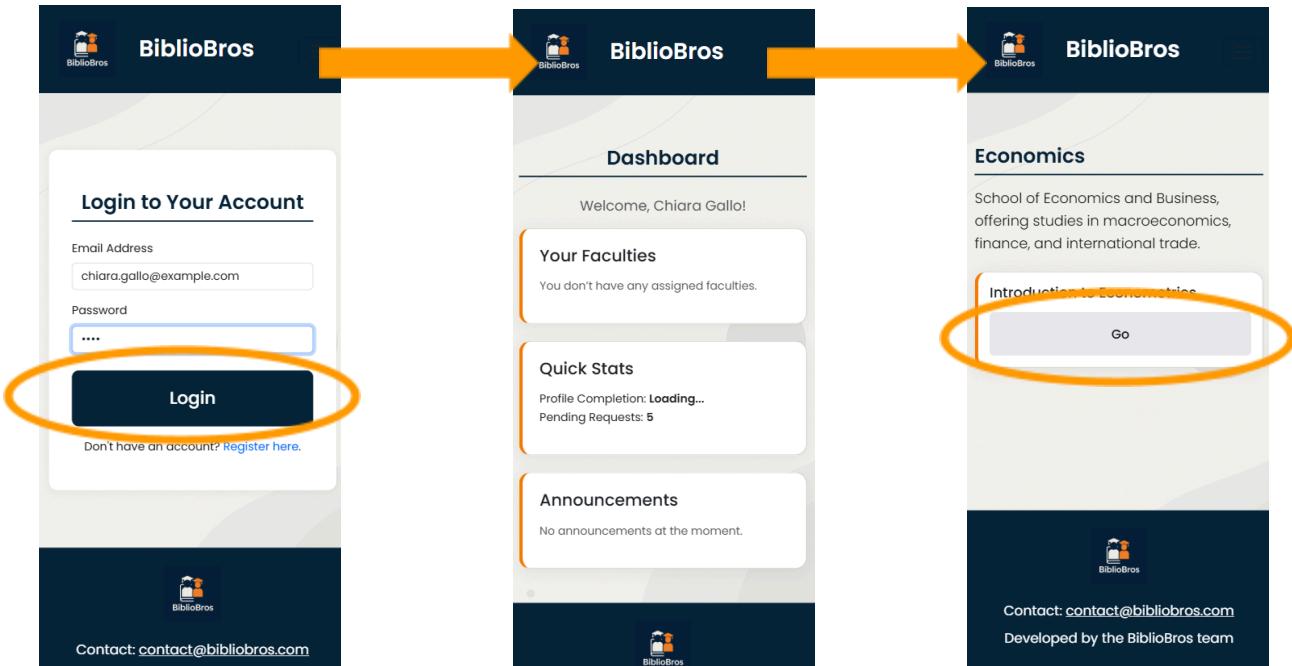
Design and Implementation

In this section, we focus on the design and implementation of the web interface. Unlike traditional approaches that rely on wireframes or mockups in the early stages, we chose to work directly on the front-end, using HTML and CSS to build and visualize the actual interface layout from the start. This allowed us to immediately observe the structure, styling, and navigation flow of the platform, making adjustments in real-time as the interaction logic evolved.

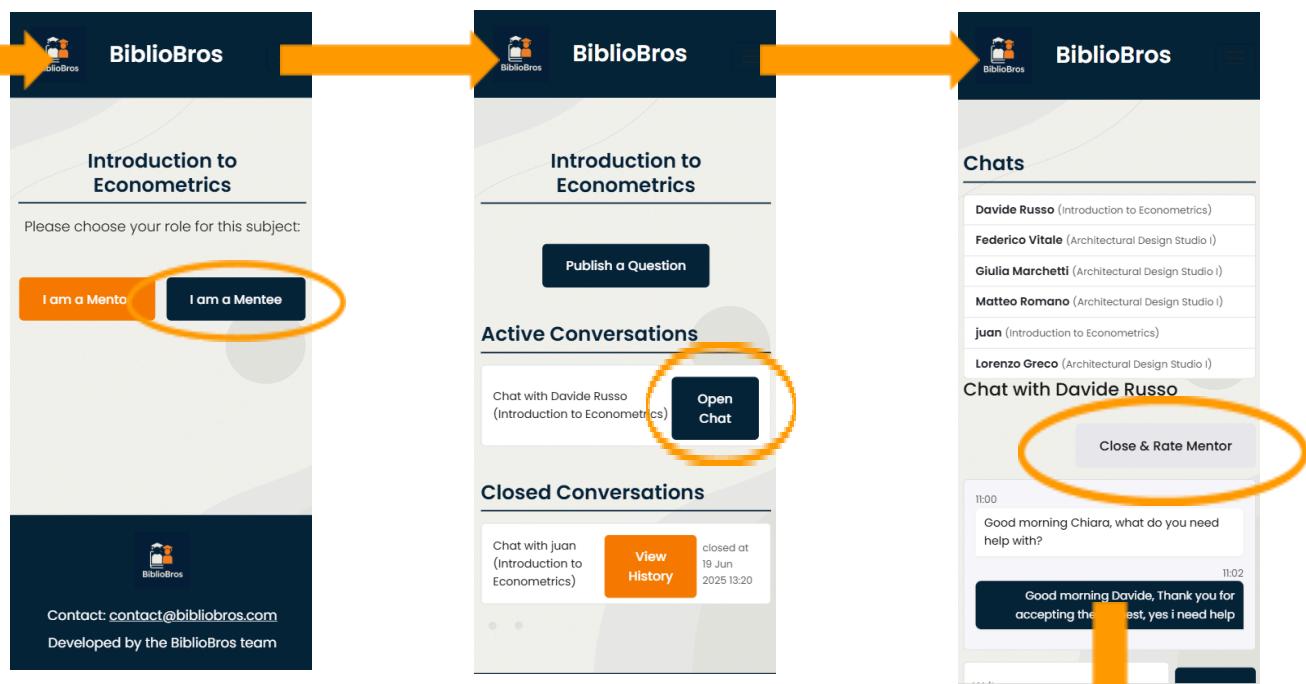
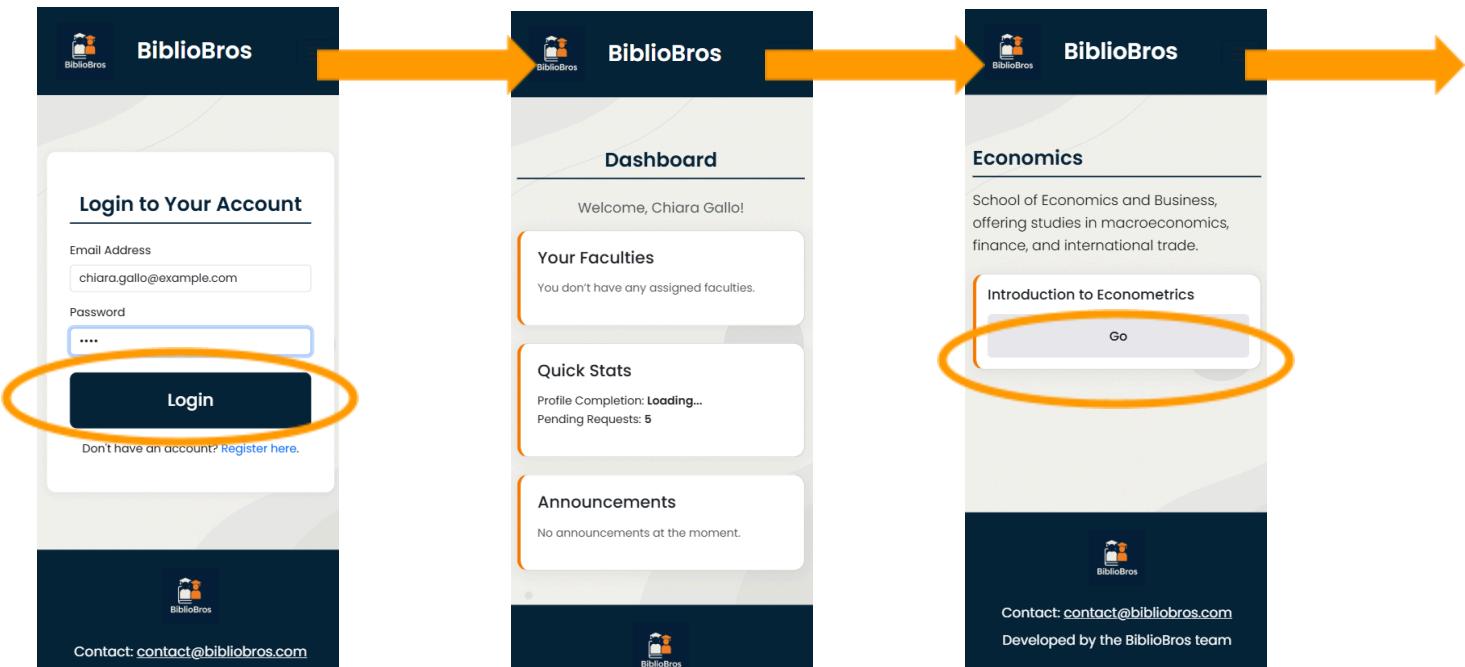
At this stage, the development remains focused on the visual and navigational aspects of the system. The full integration with a back-end and database will be completed during the prototyping phase. However, for the purposes of this phase, the static front-end already allows us to simulate the main use cases and follow the flows defined in the HTA and STN diagrams presented earlier.

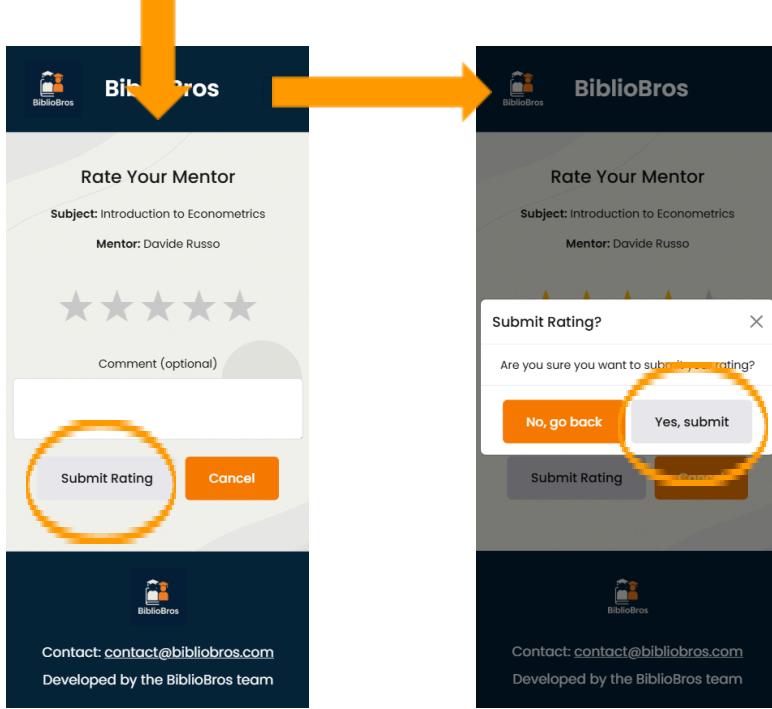
The following subsections provide a detailed walkthrough of the interface, illustrating each task flow with screenshots. For every use case (publishing a help request, rating a mentor, accepting a request, and viewing a closed chat), we demonstrate how the current version of the interface supports the user actions defined in the task models.

Make a Help Request

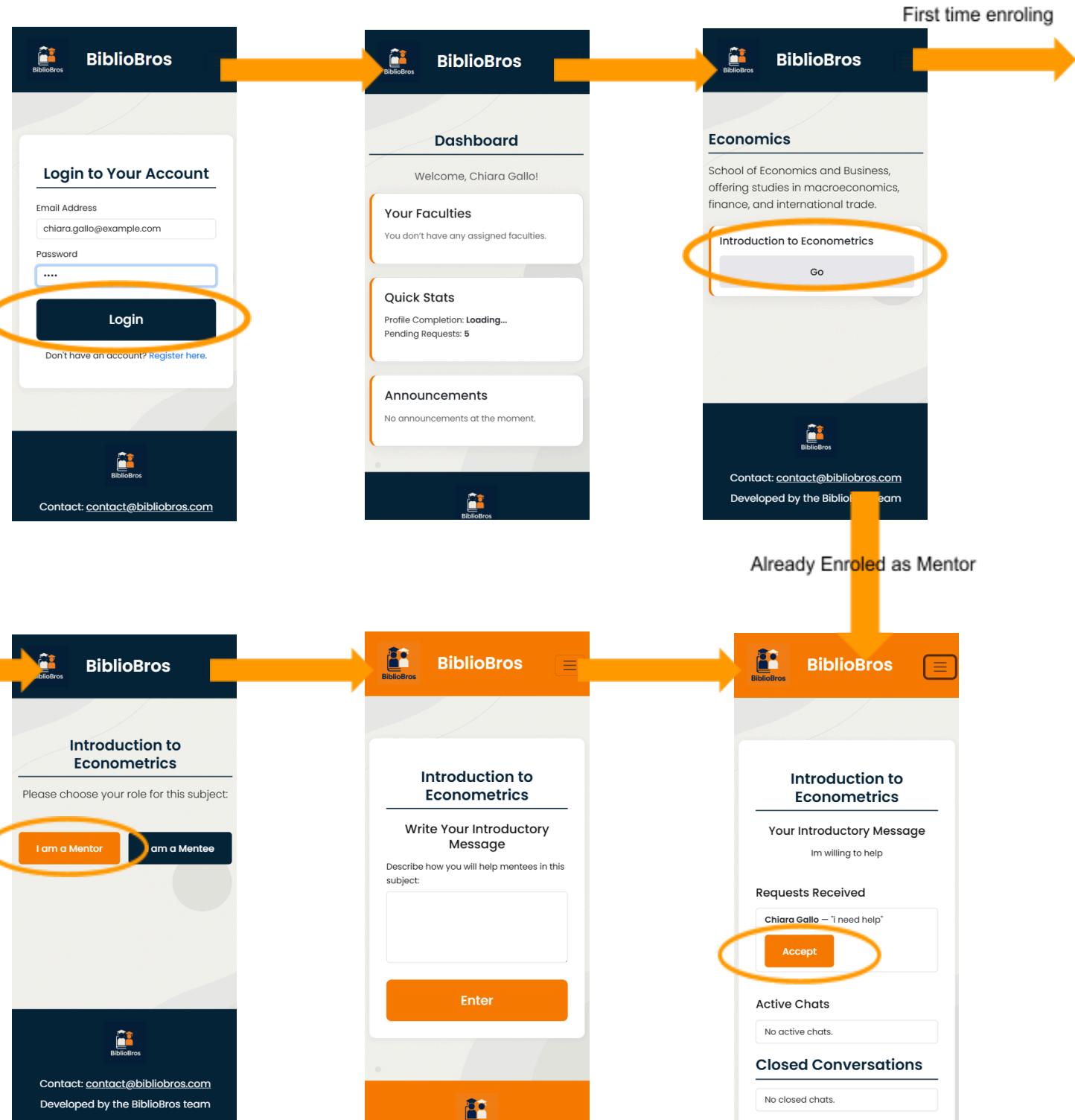


Rate a mentor

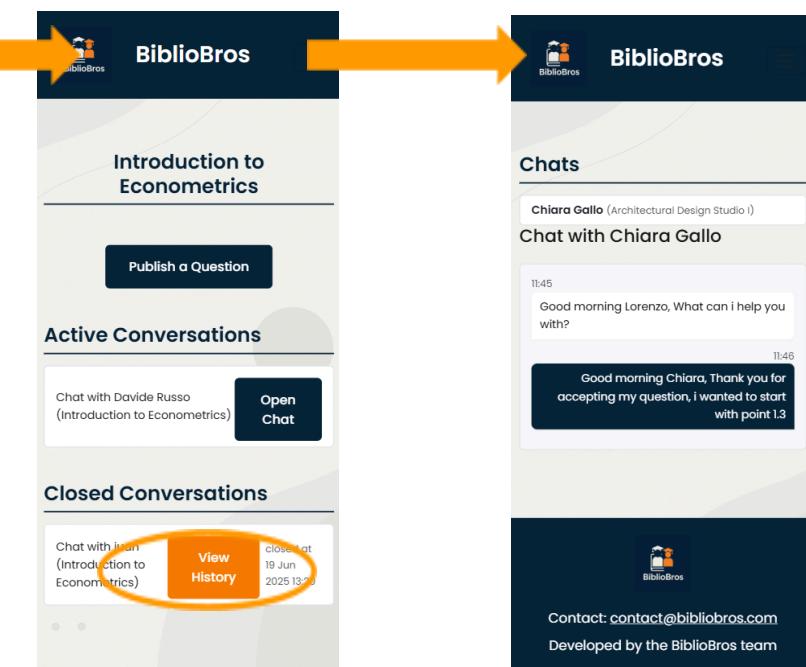
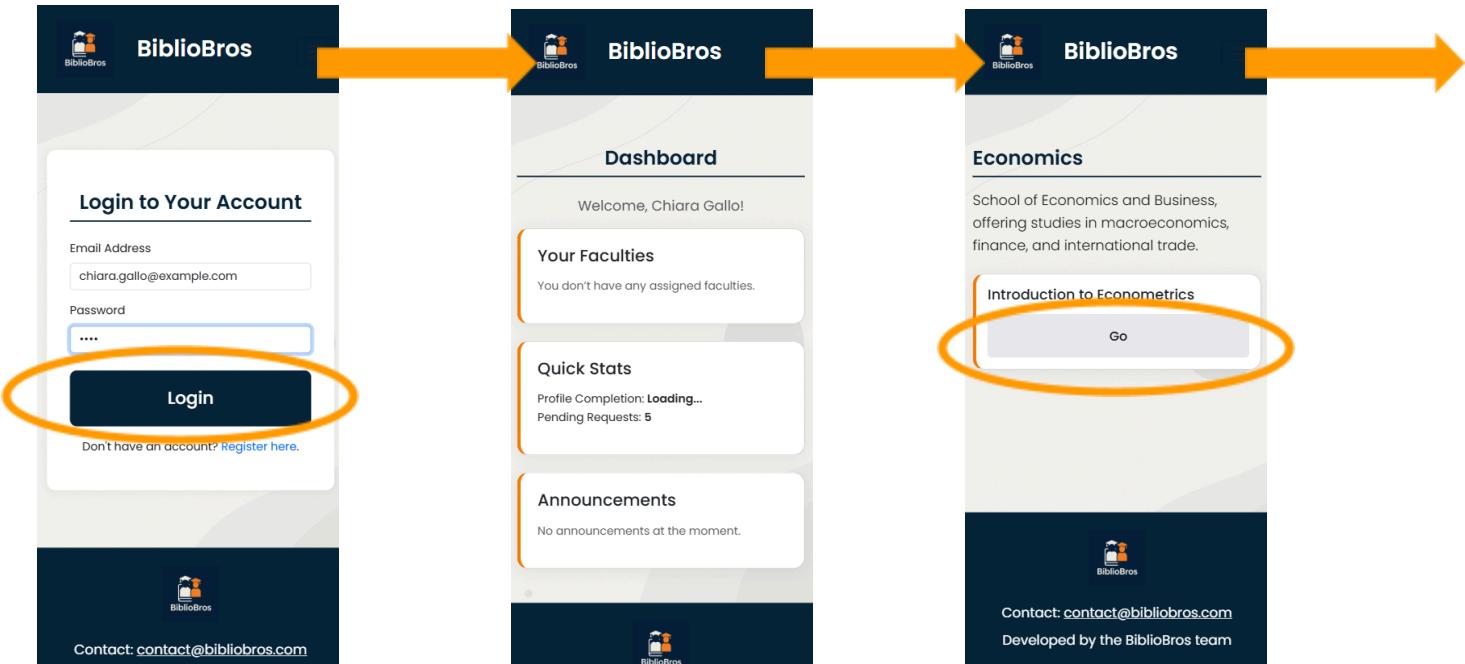




Accept a help request



View closed chats (1) Through subject



View closed chats (2) Through profile

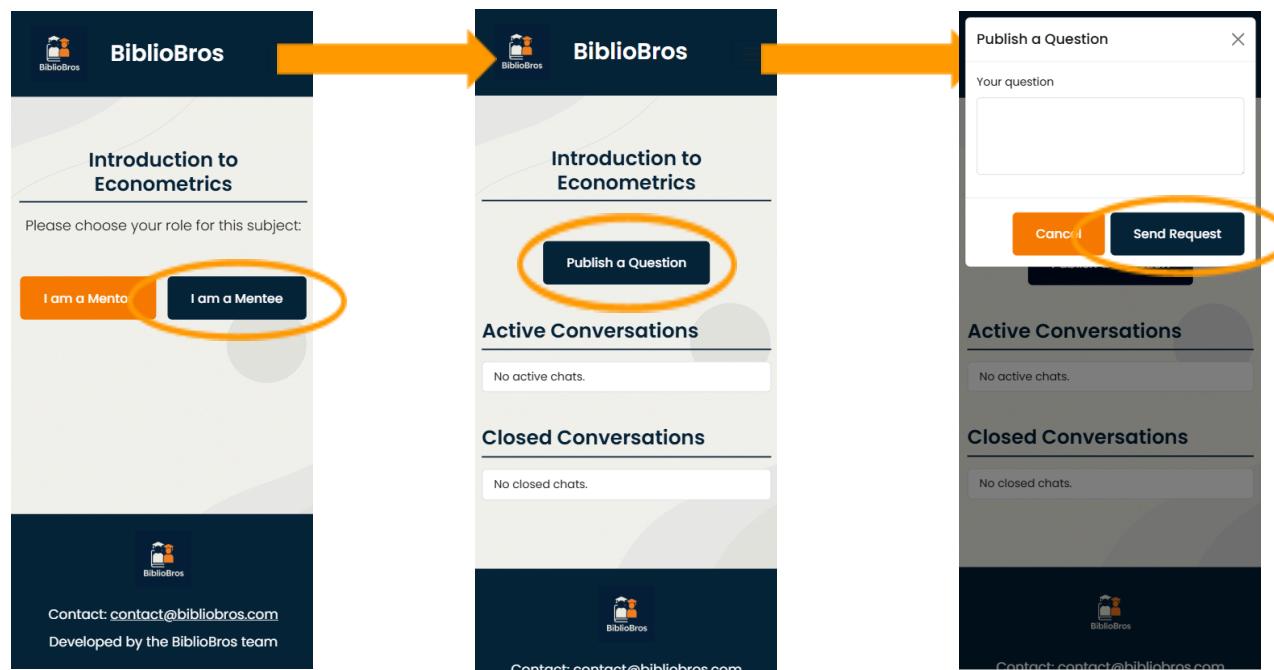
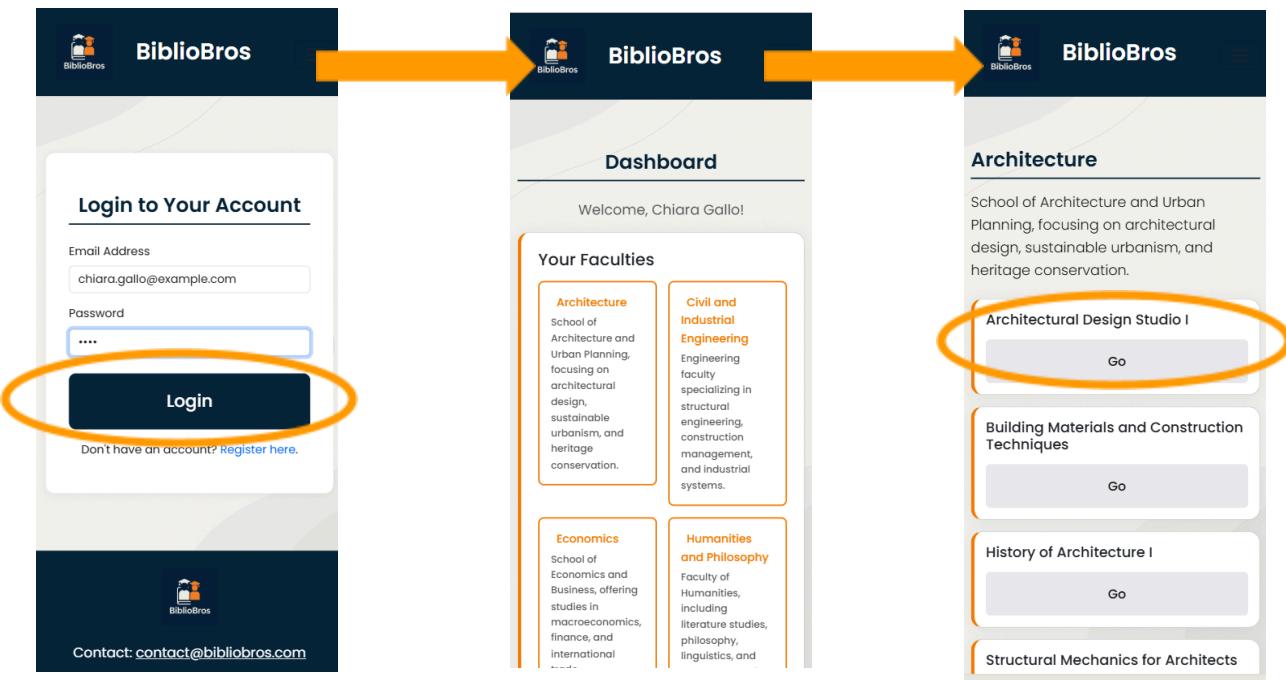


Prototype 1

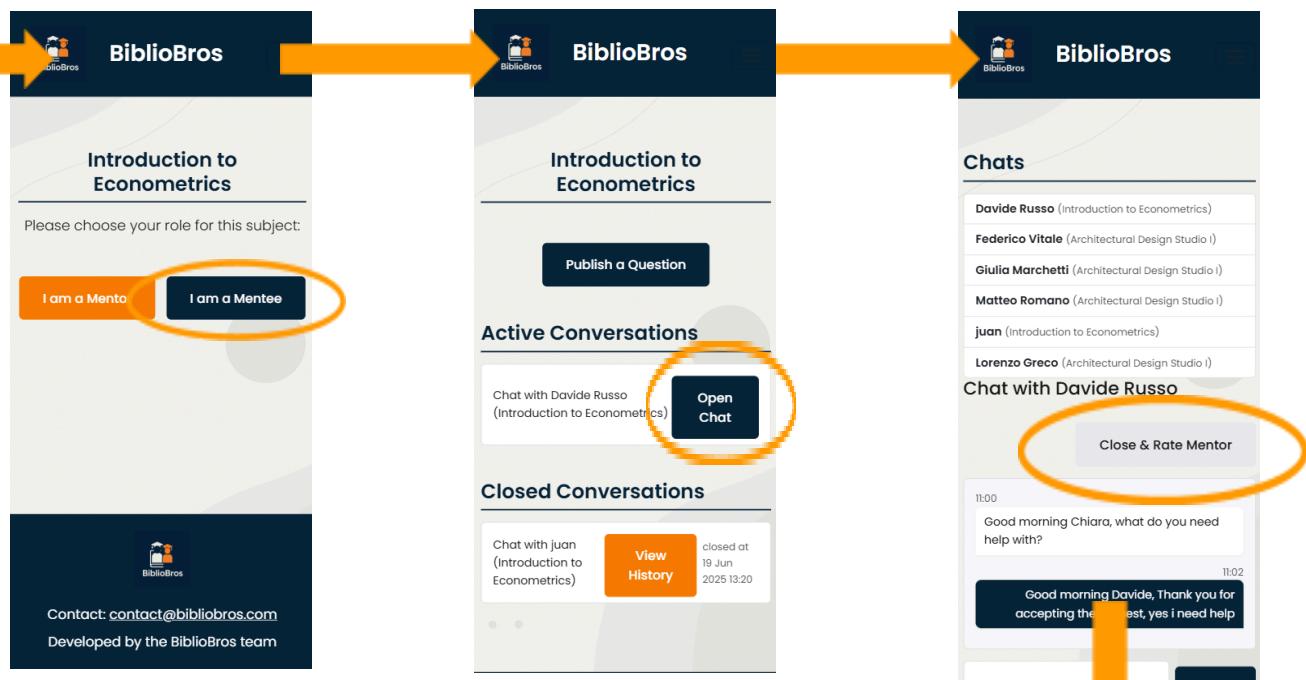
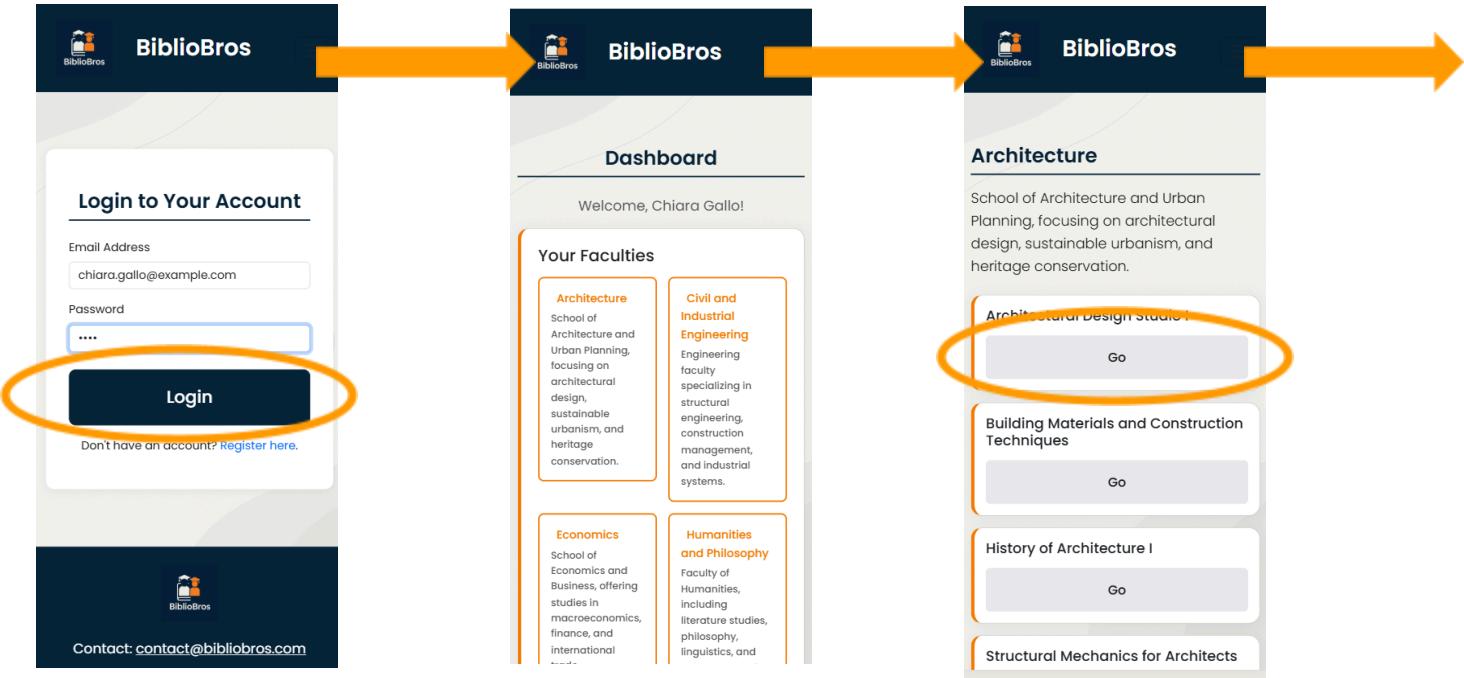
This section describes the transition from the static front-end implementation to a fully functional prototype. Unlike the previous design and implementation phase, which focused solely on the visual and structural aspects of the platform, the current version integrates a working backend and a connected database, enabling full interaction with dynamic content and persistent user data.

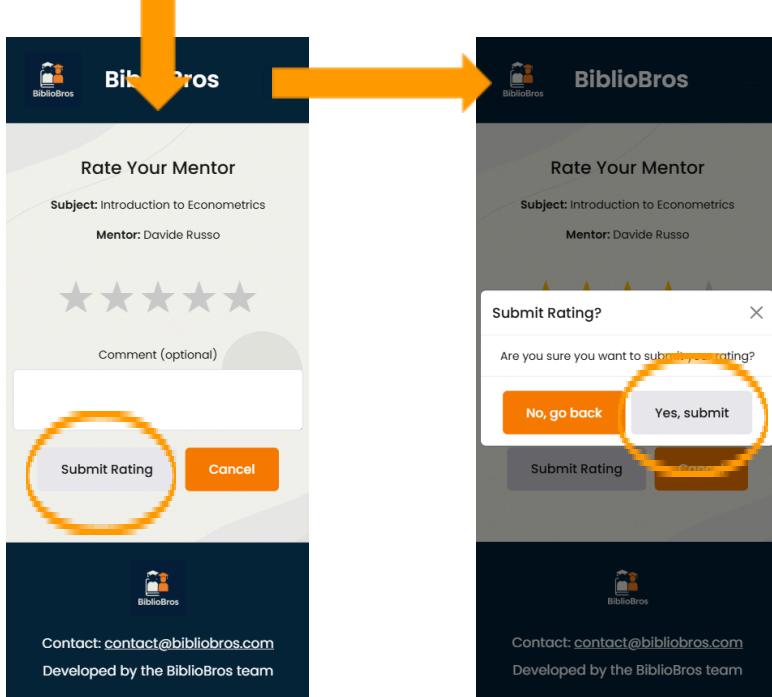
Thanks to this integration, the web application now supports real-time operations such as account authentication, request submission, chat activation, and feedback storage. This evolution significantly improves the user experience and system coherence, making the interface appear more complete, responsive, and reliable. Screens, interactions, and flows presented in this version now reflect the actual behavior of the system under real usage conditions.

Make a Help Request



Rate a mentor





Accept a help request



View closed chats (1)



View closed chats (2)



Final Realization

The final implementation of *BiblioBros* successfully delivers all the essential functionalities originally envisioned for the platform. Students can register and authenticate, publish help requests by selecting subjects and writing messages, accept incoming requests as mentors, initiate and maintain chat conversations, close the session, and rate the experience.

This version stands out from earlier stages due to the integration of a fully functional backend and database, which enabled the transition from a static HTML/CSS prototype to a dynamic, interactive web application. Thanks to this, the system now supports data persistence and a consistent user experience across sessions. Navigation is clear and task flows follow the logic defined in the HTA and STN diagrams, ensuring that each interaction aligns with the intended user behavior.

Among the strengths of the system are its clean and intuitive interface, the direct mapping of task models to real workflows, and the modular code structure, which would allow future extensions. Navigation is designed to be simple and efficient, keeping user focus on the task rather than the interface.

However, some limitations persist. The chat feature, while functional, does not support real-time updates. Additionally, user management is limited to the mentee/mentor roles, and no administrative backend or role-specific dashboard has been implemented. These aspects were deprioritized to focus on core interaction flows and delivering a stable, testable product within the available time and team resources.

Conclusions and future work

The development of *BiblioBros* has resulted in a fully functional and coherent platform that meets the core goals defined at the outset of the project. Through progressive iterations and integration of both front-end and back-end technologies, the system now enables structured peer-to-peer academic support in a simple and intuitive way. The adoption of task-based design methods such as HTA and STN proved particularly valuable in shaping a user-centric interface, ensuring clarity and consistency across different workflows.

Throughout the process, several challenges emerged—particularly related to balancing development priorities within time constraints. Despite this, the implementation was completed with a working architecture, a clean and responsive interface, and persistent data flow between users and system components.

Looking ahead, there are several opportunities for further enhancement. In particular, the "Quick Stats" and "Announcements" sections on the dashboard, although present in the interface, are currently static or underdeveloped. These areas could be expanded with real-time metrics on user activity (e.g., number of sessions completed, mentor response rate, performance insights) and platform-wide notifications (e.g., maintenance alerts, university-wide academic events, or system updates). Adding these features would improve user engagement, provide immediate feedback on system usage, and support communication across the platform.

Moreover, implementing live chat capabilities, advanced filtering in request listings, and mentor reputation indicators would further strengthen the application's value. These improvements, along with continued usability testing and feedback collection, form the natural next steps in the evolution of *BiblioBros*.