Campus Exchange A Second-hand Trading App

CS673 Software Engineering Iteration 0

Team2: Scrumble Bug

Date: Sep 26 2024

Team Building



QA Leader



Yueyihan Qi Security Leader



Yingtong Zhou Team Leader



Yuanbin Man Requirement Leader



Ang Li
Design and
Implementation Leader



HungHsu Chen Configuration Leader

Project Vision

Goal

Create a peer-to-peer platform for BU students to buy and sell used items within their campus.

Target Audience

Boston University students, including both buyers and sellers.

Problem Statement

Students have used items they no longer need, but current marketplace apps are either too broad or unsafe for campus use.



Competitor Analysis

Facebook Marketplace

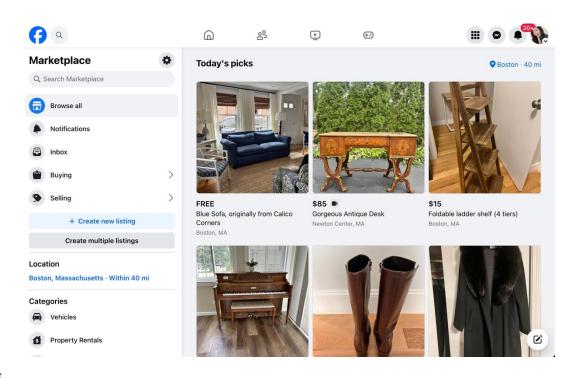
A peer-to-peer platform that provides second-hand product transactions and offline services.

Limitations

- Unrelated search results (too broad)
- Unrelated recommendations
- Lack of comments on listings

How is Campus Exchange different from Facebook Marketplace?

- Al-powered search, results are related to BU students
- Include comments on product listings that enhances UX



Competitor Analysis

WeChat Group

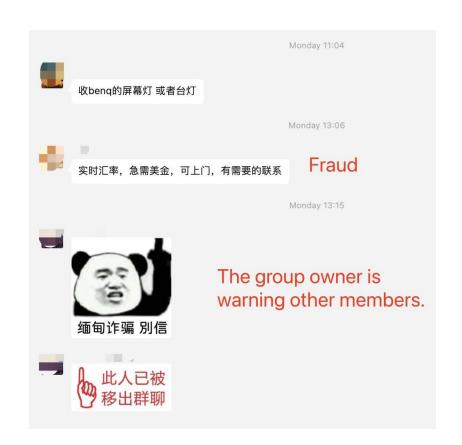
Groups of strangers posting unused items.

Limitations

- Spam and fake listings
- No search and filter functions

How is Campus Exchange different from WeChat Group?

- Secure user registration/authentication
- Large-scale database and cache system for fast data access



Milestones and Iteration Plan

Iteration 1 – Oct 17 2024

- MVP Development (user registration, item listing, comments)
- Stable database and cache system configuration

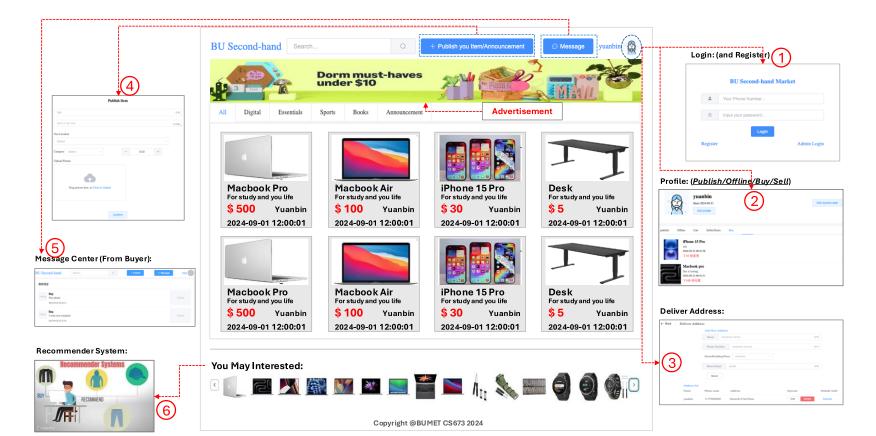
Iteration 2 – Nov 7 2024

- Enhanced Features (chat, geolocation, Al search and recommend)
- QA plan and continuous testing

Iteration 3 – Dec 5 2024

Post-launch feedback collection from students

Product Prototype



Cloud-Based Infrastructure

Alibaba Kubenetes

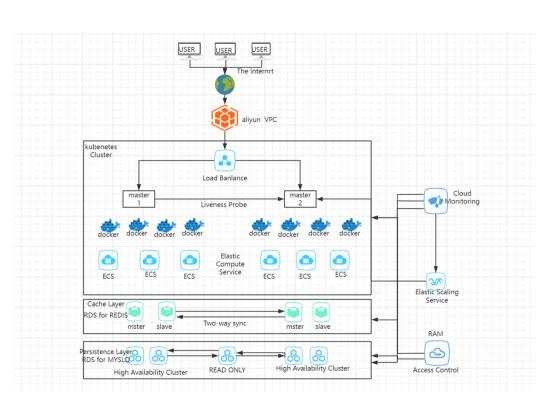
Build service architecture

Redis

Distributed database for fast data access

RDS for MySQL

Automatic backups, failure recovery



Tech Stack

Frontend Vue.js



DevOps/CI/CD
Docker
Kubernetes





Backend
Sprint Boot MyBatis





Middleware Kafka



Database MySQL



QA Testing PTS Jenkins



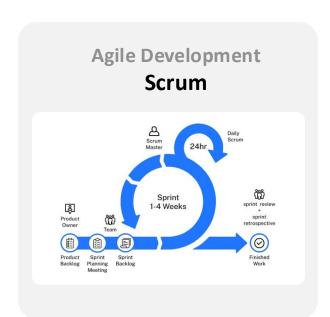
AI LLaMA

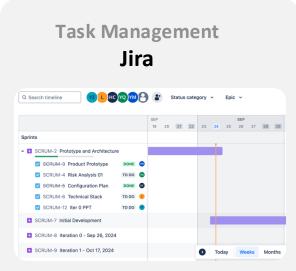


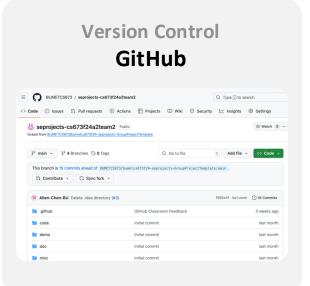
Caching Redis



Development Methodology







Risk Management

Communication

- Culture and language
- Proper documentation and weekly scrum meeting

Technical Risks

- Scalability, Data flexibility, System stability
- PTS, unit tests, integration tests, GitHub Action
 Tester and user test

Safety Risks

- Personal data
- User authentication

Thank you!