1. Software Design as an artifact refers to producing tangible outputs of design. We come up before or during development of the software with these things. For example, outputs like wireframes, architecture designing .This involves coming up with the design systems like fonts, typography and consistent colors of the platform we want to build. We also come up with architecture designs of the platform like with the USEP; we come up with diagrams how different APIs are going to communicate. We also include documentation of the maintenance plans for the USEP.This helps in having a clear vision of how the platform is going to work or function.

   Software design as a process involves defining and identifying a problem and solving problem. It involves steps like problem definition and research for example through surveys and constraint analysis by looking at the budget before diving into design of the platform. For example, for the USEP platform we can do a survey even on international students to help understand better, what would look like a success for the platform if it were created. It involves gathering information from the school management on which notifications they deem to be important which should be taken into consideration .It then involves coming up with how the system should be structured for example deciding the architecture we are going to use and the cons and pros for using such an architecture for the platform we want to build.

**Artifacts we might consider for the platform:**

1. Interactive prototypes- An early working version of the platform will help us to test early on how the students can actually use the platform for example international students who have diverse needs. For example we can easily test to see if the platform is able to handle different timezones and if not we can easily fix the issues before creating a full platform since the budget is also limited because maintenance cost would be higher.
2. Simple System Architecture Diagram
   This helps to show the development team specifically the backend on API endpoints, data structures and technical flow .This prevents costly revisions as well.

   .Trends and how they apply in USEP

   Software design trend is an emerging way of building a program at a given time. It is expected to create a platform that is designed around users' needs and this include cultural inclusivity, diverse language and accessibility. For the USEP platform, we can implement trends like modular and cloud native architecture, AI –augmented design and human –centered and inclusive design.

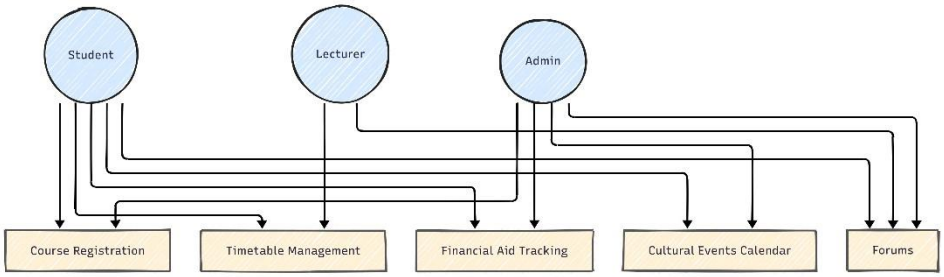
   For the platform, we have to create the platform keeping in mind that we have to suit all students,that is , local as well as international so that there is no discrimination .So as a result , we have to research on the students so that the platform is accommodating to then as wells so we have to implement human-centered design and inclusive design. For example , we take into consideration on whether international students can navigate through the portal. We also look at whether students with disabilities can easily access the platform without barriers. We need to ensure equity and inclusion is spread across for every student. For example keyboard navigation , alternatives for images and screen readers.

3. UML diagrams: Modeling interactions, workflows, and structures.

   • Architectural Decision Records (ADRs): Documenting why certain choices (e.g., microservices over monolithic) were made.

   • Pipeline diagrams: Showing CI/CD and DevSecOps workflows.

Relevant Artifacts for USEP




1. UML Use Case Diagram (Student Interactions)
    - Register for Courses
    - View Timetable Check Exam Results
    - Consult AI Advisor
    - Track Financial Aid Participate in Clubs/Events

Modular and cloud native design is another trend, which can be implemented for the platform.

Modular design is whereby we break down the entire system into smaller manageable parts (modules) making it easier to manage the system. For example for the platform we can have modules like we can have a course and registration module that only focuses on course registration by students, a communication and notification module which is only responsible for alerts ,reminders of upcoming events at the institution and payment module to keep track of invoices and tuition payments.

We can also implement cloud native design whereby the USEP platform is built in a way that it sits in the cloud. For the USEP platform, we can host the platform on a cloud server like AWS so that the student can access the platform at anytime. This scales well even to an increasing number of students. This prevents the platform from lagging because of an increase in the number of students using the platform.

Another trend is AI-Augmented Design .This is the use of Artificial Intelligence to assist in creating, testing software rather than doing it manually. This helps in saving time and improve accessibility. Implementing AI to platform development would mean that prototyping is made easier as well.AI can suggest improvements for multi-language support as well.

2. Business Case for USEP

**Executive Summary**

The current system is disjointed. Student services such as course registration, access to grades and transcript as well as class schedules and notifications are scattered across many separate systems that do not communicate with each other. As a result, students have to access different platforms based on the services they want which proves to be confusing for many students and is also time consuming. This also results in students missing important information and deadlines as messages are sent in different platforms. The Unified Student Experience Platform seeks to integrate the services into a single platform that is easier to navigate for the students.

**Problem Statement**

Students are now navigate services through multiple systems to access different services leading to problems. The problems include:

*Lower student retention*

Students disengage and stop using the platform because of the need to navigate into various systems just to do tasks or services like checking grades, registering for courses, viewing transcript in addition, checking for updates from the school. Because of accessing different portals, students miss important information like deadlines and feel disengaged.

*Operational Inefficiency*

Scattered services means that the school is spending a lot just for these different platforms to operate. More money is being spent as well on maintaining this different platforms rather than integrating it into a single platform. This is because of each platform having its own maintenance. Each platform also has its own subscription as well which adds more to the costs.

*Increased security vulnerabilities and compliance risk*

Each separate system has its own login system as well as permission. Having separate systems increases security vulnerabilities and it becomes difficult to meet compliance laws such as data protection laws when data is scattered across multiple vendors.

**Expected Benefits**

- *Operational efficiency*
    - Administration burden is reduced through automated workflows as well as data integration, which the USEP will offer. There is also elimination of the same data entry across multiple systems.

- *Student Retention*
    - By having one centralized platform, there is no need for students to navigate across different systems. Students are likely to engage more. They are most likely to complete tasks like course registration and seeking academic support on time, reducing cost, improve efficiency and getting new ideas. Students are also able to keep track of deadlines like course registration, hence low dropout rates.

3. Outsourcing is the practice of hiring another company to handle a specific task for example doing the backend of a project for another company. Different types of outsourcing includes onshore, nearshore and offshore outsourcing.

Nearshore outsourcing is whereby a company hires a service provider in a neighboring country. It helps in cost reduction while keeping communication and collaboration easier. This allows the development team to respond quickly to platform updates as well. With nearshore outsourcing, the advantage is that people likely have similar languages and cultures thus reducing misunderstandings. Limited talents are discovered as neighbouring countries may have fewer specialist for example AI specialists. Nearshore outsourcing is not ideal because different regulatory laws. For example laws on different on data security.

Offshore outsourcing is a practice whereby a company delegates service provider in a distant country. For example, a country in Zambia delegates a service provider in Australia. Offshore is done to reduce cost. For example, a company in India delegates a company in Zambia to do the backend of the project where labour costs are cheaper. There is access to a larger pool of talented people as compared to nearshore outsourcing. Offshore provides many skilled AI experts and their ideas and insights help in making a better system, for example experienced data analyst to improve student-matching features The con of offshore outsourcing is that communication can prove to be a challenge. Language and cultures may cause misunderstandings. Time zone difference as well can cause delay between the company and offshore developers. Another problem with offshore outsourcing is difference in laws and regulations for example labour laws and data protection. For example, working hours differ by country, therefore a need to adjust the contract`s conditions to match local labour regulations.

Onshore outsourcing is when a company hires a service provider within the same country. This is done to improve efficiency, maintain quality and ensuring regulatory compliance.

The advantages of onshore outsourcing includes better control as well as easy communication. As people who are within the same country are working together, language is not a barrier and time zones are the same so the company can work well with the service providers.

The disadvantages with onshore outsourcing is that expanding or limiting the team may be limited by local labour availability and certain specialist may be harder to find.

In conclusion, the best outsourcing approach is offshore outsourcing considering a platform like the USEP .This is because it is also meant to cater for international students .Working with an offshore team allows us to get talent from different countries making the app more inclusive and developing a platform/ system that put into mind cultural differences.

**Culture & Ops Lead Responses**

**1. Cultural Intelligence**

The USEP must reflect the diversity of an international student body. Two key inclusivity requirements: 1. **Multilingual UI**:

- English as the default, but also support local Zambian languages (e.g., Bemba, Nyanja) and global languages (French, Chinese, Arabic).
- This reduces barriers for international and regional students.

2. **Accessibility Features**:
   - Screen reader compatibility, high contrast mode, and adjustable text sizes.
   - Optimized for low-bandwidth users (lightweight mobile app, offline caching). o Ensures students with disabilities or poor internet still access services.

Bottom line: **A platform that ignores inclusivity risks alienating the very students it is designed to support.**

---

**2. DevOps & DevSecOps • Pipeline**

**Flow (CI/CD)**:

- Code → Build → Test → Security Scan → Deploy → Monitor • **Benefits**:
  - Faster, more reliable updates. o Continuous security scanning prevents vulnerabilities from slipping into production.
  - Monitoring ensures the platform scales sustainably as enrollment grows.
- **Sustainability link**: automated pipelines mean fewer manual errors, less downtime, and better use of resources (both financial and environmental).

---

**3. AI & Ethics**

- **Opportunity**:

    - AI powered **academic advising chatbot**: helps students plan courses, reminds them of deadlines, and provides quick answers 24/7.
    - AI based **financial aid assistant**: tracks applications, alerts on loan repayments.
- **Ethical Concern**:
    - **Bias in AI**: if trained on limited data, it may favor certain student groups or recommend "easier" paths that limit opportunities.
    - **Privacy**: misuse of student data if governance is weak.
- **Approach**: AI must be **transparent, explainable, and human supervised**.