Library Management System - Project Part 1: Vision/Plan

EECS447: Database Systems
Team Name: ASYNClib
Team Members: Nick, Ashton, Cole, Sean, Yadhunath

Professor: Hossein Saiedian
Semester: Fall 2025

Revisions

Version: 1.3

Version	Date	Author	Description
1.1	08/27/25	Nick, Ashton, Cole, Sean, Yadhunath	Made the initial draft, assigned roles, created the document template
1.2	09/03/25	Nick, Ashton, Cole, Sean, Yadhunath	Worked on the Vision Statement, Scope Statement and team organization and profile.
1.3	09/10/25	Nick, Ashton, Cole, Sean, Yadhunath	Made some final changes to the document.

Vision Statement:

Libraries stand as epicenters of knowledge, holding vast collections of data within limited physical space. Managing this wealth of information effectively can be a daunting challenge. Database systems offer a powerful solution by transforming complex data management into something simple and efficient — just a few keystrokes away.

At ASYNCLib (named after the initials of our team members), our vision is to make library data an accessible resource rather than a burden. We aim to design a clear and comprehensive database that organizes essential information about books and resources, while offering features that make access simple, intuitive, and enjoyable for every user.

We want to create an experience where clients can quickly find what they need, explore collections with ease, and interact with the system without technical barriers. At the same time, we are committed to maintaining strong safeguards, including secure and anonymous views, to ensure user privacy. Ultimately, our goal is to reimagine library data management — turning it into a tool that empowers, supports learning, and makes the library experience smoother for everyone.

Scope Statement:

Our scope of the project would extend, and is not only limited to the following:

❖ *A resource management system*

   ➢ Books will include attributes such as title, author, ISBN, publication year, genre, and availability.
   ➢ Magazines will include attributes such as title, issue number, publication date, and availability.

❖ *A client and membership management system*

   ➢ Clients will have attributes such as unique ID, name, contact information, membership type, and any borrowing restriction
   ➢ Multiple membership categories will exist such as regular, student and senior

❖ A system for tracking borrowed, returned and reserved media

   ➢ A system for users to reserve media currently on loan
   ➢ All media will be tracked with timestamps and client details

- Enforce constraints on borrowed items depending on membership type
- Enforce constraints on late return fees depending on membership type

❖ A user interface

- ➢ Will allow Library Staff to check out items, process returns, add new items, and manage client accounts.
- ➢ Will allow for generation of fine calculations, media availability status, and client profiles.

Team Organization and Profiles:

❖ **Ashton**
- ➢ ***ROLE***: Deployment Lead
- ➢ ***CONTACT***: a492s477@ku.edu
- ➢ ***EXPERIENCE***: 2 years Python, 1 year C++

❖ **Nick**
- ➢ ***ROLE***: Team Lead
- ➢ ***CONTACT***: Nick.Paslay@ku.edu
- ➢ ***EXPERIENCE***: 2 years python, 1 year C/C++, compilation, embedded devices

❖ **Sean Crosby**
- ➢ ***ROLE***: Quality Engineer
- ➢ ***CONTACT:*** s804c771@ku.edu
- ➢ ***EXPERIENCE:*** 1 year MySQL, 2 years JS and node.JS, 4 years Python

❖ **Cole Cooper**
- ➢ ***ROLE:*** Database Administrator
- ➢ ***CONTACT:*** c459c999@ku.edu
- ➢ ***EXPERIENCE:*** 2 years Python, 1 year C++

❖ **Yadhunath Tharakeswaran**

- ➢ ***ROLE***: Lead Data Engineer
  - Would serve responsible for constructing the database and be providing insights and making the database more scalable for the provided use case.
  - Will be collaborating alongside other team members on updates, issues, and performance of the database.
  - Would also be the main point of contact before making any major changes to the database structure and architecture.

- ➢ ***CONTACT:*** yadhu2003@ku.edu (Availability: Tu-Th → 9am-12pm)

➢ ***EXPERIENCE:*** Python (4yrs), MySQL & PostgresSQL (2yrs), MongoDB (1yr), React & Node.js (2yrs), Django (1yr).

Upcoming improvements: