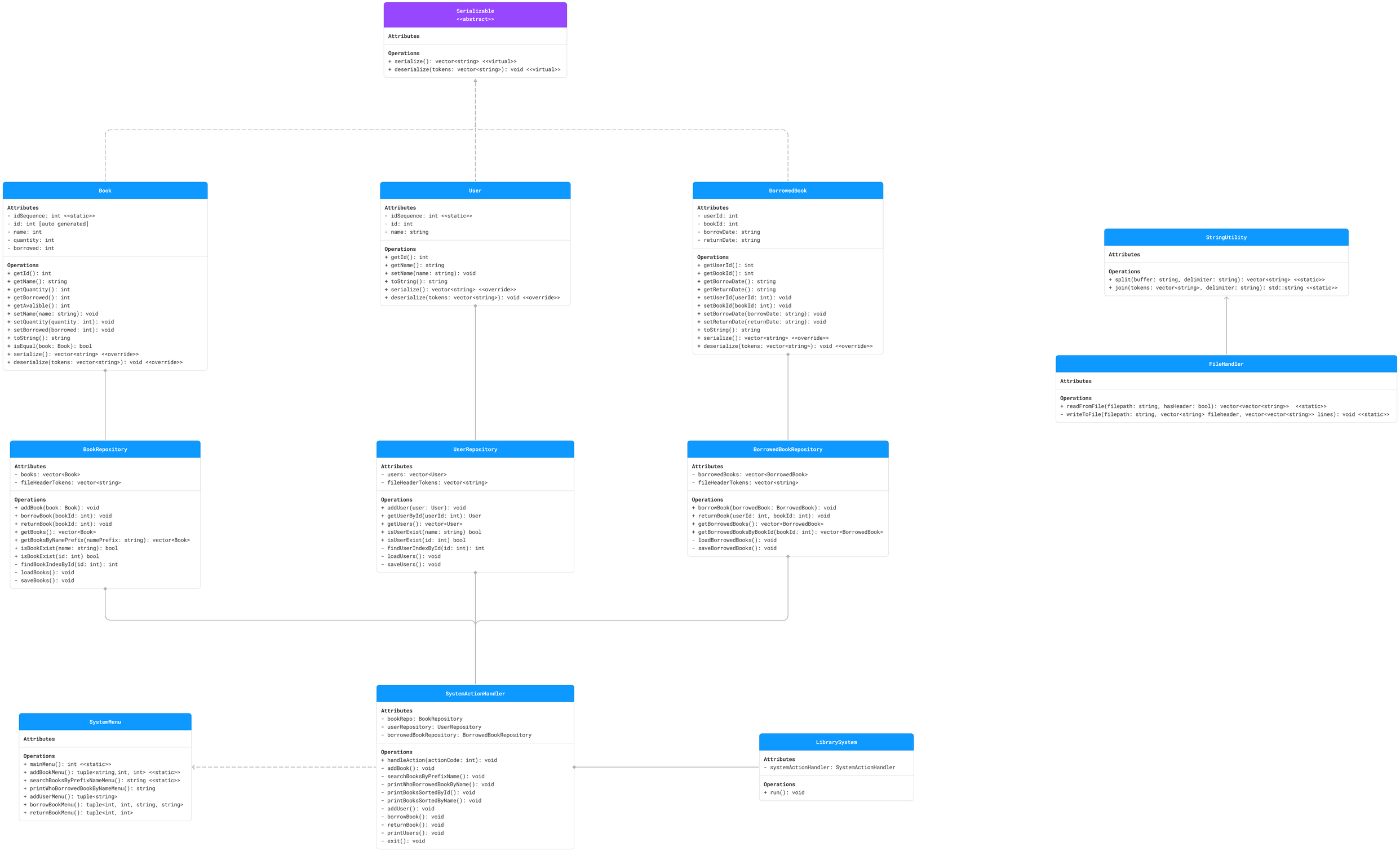
Serializable
<<abstract>>
Attributes
Operations
+ serialize(): vector<string> <<virtual>>
+ deserialize(tokens: vector<string>): void <<virtual>>
Book
Attributes
- idSequence: int <<static>>
- id: int [auto generated]
- name: int
- quantity: int
- borrowed: int
Operations
+ getId(): int
+ getName(): string
+ getQuantity(): int
+ getBorrowed(): int
+ getAvalible(): int
+ setName(name: string): void
+ setQuantity(quantity: int): void
+ setBorrowed(borrowed: int): void
+ toString(): string
+ isEqual(book: Book): bool
+ serialize(): vector<string> <<override>>
+ deserialize(tokens: vector<string>): void <<override>>
User
Attributes
- idSequence: int <<static>>
- id: int
- name: string
Operations
+ getId(): int
+ getName(): string
+ setName(name: string): void
+ toString(): string
+ serialize(): vector<string> <<override>>
+ deserialize(tokens: vector<string>): void <<override>>
BorrowedBook
Attributes
- userId: int
- bookId: int
- borrowDate: string
- returnDate: string
Operations
+ getUserId(): int
+ getBookId(): int
+ getBorrowDate(): string
+ getReturnDate(): string
+ setUserId(userId: int): void
+ setBookId(bookId: int): void
+ setBorrowDate(borrowDate: string): void
+ setReturnDate(returnDate: string): void
+ toString(): string
+ serialize(): vector<string> <<override>>
+ deserialize(tokens: vector<string>): void <<override>>
StringUtility
Attributes
Operations
+ split(buffer: string, delimiter: string): vector<string> <<static>>
+ join(tokens: vector<string>, delimiter: string): std::string <<static>>
FileHandler
Attributes
Operations
+ readFromFile(filepath: string, hasHeader: bool): vector<vector<string>> <<static>>
- writeToFile(filepath: string, vector<string> fileheader, vector<vector<string>> lines): void <<static>>
BookRepository
Attributes
- books: vector<Book>
- fileHeaderTokens: vector<string>
Operations
+ addBook(book: Book): void
+ borrowBook(bookId: int): void
+ returnBook(bookId: int): void
+ getBooks(): vector<Book>
+ getBooksByNamePrefix(namePrefix: string): vector<Book>
+ isBookExist(name: string): bool
+ isBookExist(id: int) bool
- findBookIndexById(id: int): int
- loadBooks(): void
- saveBooks(): void
UserRepository
Attributes
- users: vector<User>
- fileHeaderTokens: vector<string>
Operations
+ addUser(user: User): void
+ getUserById(userId: int): User
+ getUsers(): vector<User>
+ isUserExist(name: string) bool
+ isUserExist(id: int) bool
- findUserIndexById(id: int): int
- loadUsers(): void
- saveUsers(): void
BorrowedBookRepository
Attributes
- borrowedBooks: vector<BorrowedBook>
- fileHeaderTokens: vector<string>
Operations
+ borrowBook(borrowedBook: BorrowedBook): void
+ returnBook(userId: int, bookId: int): void
+ getBorrowedBooks(): vector<BorrowedBook>
+ getBorrowedBooksByBookId(bookId: int): vector<BorrowedBook>
- loadBorrowedBooks(): void
- saveBorrowedBooks(): void
SystemMenu
Attributes
Operations
+ mainMenu(): int <<static>>
+ addBookMenu(): tuple<string,int, int> <<static>>
+ searchBooksByPrefixNameMenu(): string <<static>>
+ printWhoBorrowedBookByNameMenu(): string
+ addUserMenu(): tuple<string>
+ borrowBookMenu(): tuple<int, int, string, string>
+ returnBookMenu(): tuple<int, int>
SystemActionHandler
Attributes
- bookRepo: BookRepository
- userRepository: UserRepository
- borrowedBookRepository: BorrowedBookRepository
Operations
+ handleAction(actionCode: int): void
- addBook(): void
- searchBooksByPrefixName(): void
- printWhoBorrowedBookByName(): void
- printBooksSortedById(): void
- printBooksSortedByName(): void
- addUser(): void
- borrowBook(): void
- returnBook(): void
- printUsers(): void
- exit(): void
LibrarySystem
Attributes
- systemActionHandler: SystemActionHandler
Operations
+ run(): void