Flow of Events

Actor Action System Response

1. This use case begins when a Professor submits a grade sheet for a Section.

2. The Professor provides the professor’s identifier, the department code, course number, and section number for the section. For each Student in the Section, the Professor provides the student’s identifier and the student’s grade.

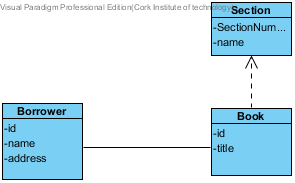
3. Records each student’s grade.

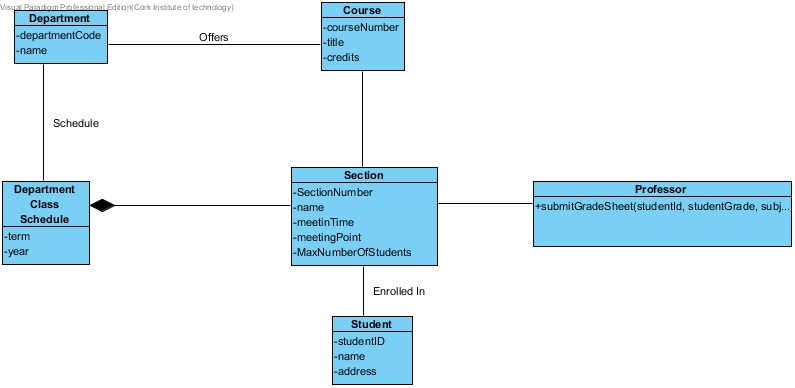
4. On completion of entry of the grades, the Professor indicates that the entry of grades is complete.

5. Produces professor’s grade report.

6. The Professor receives the instructor’s grade report

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Event Number** | **Event Description** | **System Input** | **Actor Providing Input** | **System Output** | **Actor Receiving Output** |
| 1 | Borrow book | Book ID, PersonID, Date | Borrower | receipt | Borrower |
| 2 | Return book | Book ID, Date | Borrower | Late fees | Librarian |
| 3 | Search for book | Book name/id | Borrower | Search results | Borrower |
| 4 | Pay late fees | Amount | Borrower |  | Librarian |





Operations contract

Name: enterCardNumber(cardNumber)

Responsibilities: Receive and validate card number from Patron

Cross Reference:

Exceptions: Invalid cardNumber

Preconditions:

Post Conditions:

name : enterBookIdentifier(bookIdentifier)

Responsibilities: Receive and validate card identifier from Patron

Cross Reference:

Exceptions: Invalid bookIndentifier

Preconditions:

Post Conditions: return due date to patron