**Glossary**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Description** | **Author** |
| Inception Draft | July 20, 2018 | First draft. To be refined primarily during elaboration phase. | Katy Atchison |
| Elaboration 1 Draft | July 29, 2018 | Second draft. Changes include detailing a greater amount of detail and including new terms that appeared since the last draft. | Cody Dowell |
| Elaboration 2 Draft | August 1, 2018 | Third draft. Changes include adding minor terms that were recently discovered during elaboration 1. | Cody Dowell |
| Final Draft | August 5, 2018 | Final Draft. Changes include fixing grammatical issues, adding more terms, updating and clarifying terminology, and visual changes to the document such as adding page numbers and italicizing the names of terms. | Katy Atchison |

|  |  |
| --- | --- |
| **Term** | **Definition** |
| **All Books** | A *Category* that stores a collection of books. This is where every single *Book* created by the *User* is stored. This *Category* can be used to generate a list of all of the user’s *Books* at any time. It acts as the default *Category* when the *User* adds *Books*. This is also where the user goes to find any *Books* that they have not otherwise categorized. This *Category* will be given to a user by default. |
| **Account** | An *Account* is the object a user will use to access their personal data in the System. Each *Account* contains its own *Books*, *Categories*, *Ratings*, and *Reviews*, added by the user who has access to the *Account*. An *Account* contains a unique *Email* and a *Password*. For someone to have access to the features of the application, they will first need to create an *Account* with a valid *Email* and *Password*. |
| **Author** | An attribute contained in the *Book* class. *User* enters this attribute when creating a *Book*. Represents the author of the physical real-world book that the *Book* class represents. |
| **Book** | A class representing a physical book that the user may have read or wants to read. Will have attributes, including *Title*, *Author*, *Description*, *Classification*, *BookID*, and *TotalNumberOfPages*, representing a number of pages corresponding to the physical book. A *Book* can be given a personal *Rating*, as well as an aggregate *Review*. A *Book* object can also be added to any number of different *Categories* that a user defines. All of these *Books* belong in the “*All Books*” *Category*. |
| **BookID** | A hidden attribute in the *Book* class that distinguishes the *Book* objects from one another and acts as a primary key in the *Database* holding all *Books*. A *Book’s* *BookID* is automatically set when the *Book* object is created and fulfills all other requirements of a primary key. |
| **BookKeeper** | *BookKeeper* is the name of the application. This term will collectively refer to the all of the software working to implement the functions and features of our application as a whole. |
| **Category** | Collection of references to *Book* objects. *Users* create and name these, and they are displayed in the user interface as a means of giving the user organization in navigating through their saved *Books*. A user is allowed to create as many *Categories* as they want, and they are also allowed to add as many *Books* to any given *Category* as they want. |
| **CategoryID** | A hidden attribute in the *Category* class that distinguishes the *Category* objects from one another and acts as a primary key in the *Database* holding all categories. A *Category*’s *CategoryID* is automatically set when the *Category* object is created and fulfills all other requirements of a primary key. |
| **Classification** | All books have a *Classification* of either “*Read*”, “*To-Read*”, or “*Currently-Reading*”. This affects what associations they have. Only “*Read*” books can have a *Rating* or *Review*, and only “*Currently-Reading*” books can have *Progress*. However, a *Book* can have its *Classification* changed- for instance, it can first be classified as “*Want-To-Read*” and then, when the user starts the book, the *Classification* can be changed to “*Currently-Reading*”. |
| **Database** | The *Database* is a central part of the software that is responsible for storing data for all of the *Users*, *Books*, and *Categories* created. The *Database* is updated any time an *Account* is registered, when any changes are made to a *Book*, or whenever a *Category* is created or edited. The *Database* will stay up at all times and will have preventive measures to stop total system failure in the case of the *Database* going down. |
| **Database Modification Authorization** | *Database Modification Authorization* is validation by the *Database* and by the software to ensure that any creation, edit, or deletion of information is valid before it is saved in the *Database*. This is to prevent system failures and act as a safeguard against lack of comprehension by the user. In the event of information not being valid, the information is rejected, and the user is prompted by the UI to enter valid information instead or abandon the operation they are attempting. |
| **Database Modification Authorization Request** | This is an action done by the user interface when information is entered by the user in an attempt to create, edit, or delete information in the *Database*. It sends the information to be authorized by means of the *Database Modification Authorization*, which will ensure that information entered is valid. |
| **Description** | An attribute contained in the *Book* class. *User* enters this attribute by typing it into a text box when creating a *Book*. Contains a short description of the physical real-world book that the *Book* class represents. |
| **Password** | The *Password* is a log in credential that a user will enter in order to log in, along with their *Username*. The *Password* field will display asterisks in place of characters when the *Password* is typed in, and the actual *Password* itself will be encrypted in the *Database* to protect it from any outside attacks. *Password* will have some restrictions regarding its strength so that people trying to gain access to others’ *Accounts* will not be able to guess a *Password* easily or quickly. |
| **Progress** | The user’s progress in the real world in a book they are currently reading. *User* will enter a page number, and *Progress* is calculated by dividing their page number by the *Book’s* *TotalNumberOfPages* and multiplying this decimal by one hundred. It is displayed as a percentage. *Progress* is stored in the *Database* and is able to be modified at any time, as long as the information is entered correctly i.e. the *User*’s page number is not greater than the *Book’s* *TotalNumberOfPages*. |
| **Rating** | The user’s personal rating out of ten for a book they have read. Only applies to *Book*s classified as “*Read*”. *Ratings*, once confirmed and validated, are stored in the *Database* and are available for the user to view at any time. The *Rating* for any *Book* can be changed at any time after it has been initially set. |
| **Read** | This is a classification that indicates that the user has finished reading a specific book. When a *Book’s* *Classification* is “*Read*”, the *User* will be able to enter a *Rating* and a *Review* for the *Book*. Until a *Book* is assigned the “*Read*” *Classification*, *Ratings* and *Reviews* are not available for that *Book*. |
| **Review** | The user’s personal “review” of a book they have read. Stored as plain text that the *User* enters through a form. *Reviews*, once confirmed and validated, are stored and are available for the user to view at any time. The *Review* can also be changed at any time after it is initially created, as long as the new *Review* fits the restrictions provided. |
| **Submit** | A user decides to click on a submit button following a creation, edit, or deletion of data that the user is currently storing or wants to store in the database. Anytime a submit button is activated, a change in the database is soon to follow, provided all data entered is valid and no system failures occur. |
| **Title** | An attribute contained in the *Book* class. *User* enters this attribute when creating a *Book*. Represents the title of the physical real-world book that the *Book* class represents. |
| **TotalNumberOfPages** | An attribute contained in the *Book* class. *User* enters this attribute when creating a *Book*. Represents the number of pages of the physical real-world book that the *Book* class represents. Used in calculating the *Progress* of a user in a “*Currently-Reading”* *Book*. |
| **User** | An object representing the user of the application. Used to view information and create, edit, and delete the information in it. Each *User* accesses the features of the application through a personal *Account* that the user must create with a Username and Password. |
| **UserID** | A hidden attribute in the *User* class that distinguishes the *User* objects from one another and acts as a primary key in the database holding all users. A *User*’s *UserID* is automatically set when the *User* object is created and fulfills all other requirements that a primary key needs in order to function correctly. |
| **Username** | The *Username* is a log in credential that a user will enter into the system when trying to log in, along with the user’s password. The *Username* will be shown as plain text when being typed into its appropriate log in field. Every *User* must have a unique username. |
| **+** | The + symbol is what a user will click on when trying to use any of the create functions of the application. This includes attempting to add a *Book* to their “*All Books*” category as well as adding a new *Category* for *Books* to be added to. |