|  |  |  |
| --- | --- | --- |
| **Group No:** | **REG/WD/04** | |
| **Brief description of the project** | This Online Library Management System is an Automated Library System that handles the various functions of the library. The librarian uses a computerized system to manage the library, where he / she can manage & track the daily work of the library such as issuing books, return books, collection of new books etc., Any person can become a member of the library by filling a prescribed form. In addition, the system includes detailed information of refunded books, the date on which the user must return the books and the fees associated with them.  If the user's position is an administrator, the user can generate various types of records such as lists of registered students, book list, payment details, issue and return reports. If the user's position is a member, the user can generate a report of the book's borrowing and returning history.  All these modules can help librarian to manage the library with more convenience and in a more efficient way as compared to library systems which are not computerized. This online Library Management System helps the users in better learning by providing quick access to the library system. | |
| **INDIVIDUAL DETAILS OF GROUP MEMBERS** | | |
| **Registration No** | IT20182946 | |
| **Student**  **Name** | Samarasinghe.V.M | |
| **Function(s)**  **Note**: Include the functions required to complete for **sprint 1** & **sprint 2** | **Sprint 1**  Implementing all the features of customer side payment   * Payment home page * Membership fee page * Late Payment page   **Sprint 2**  Implementing the all the features of admin side payment   * Member ship payment admin page * Late payment admin page | |
| **Sprint 1** | **Picture of interface 1** | **Brief description of interface 1** |
|  | **Purpose**: This interface will be used to define which payment customer is going to pay.  **Flow**: Customer must login to the system before choosing.  **Good practices**:   * Have a meaningful title on the page. * Simple and user-friendly interface. |
| **Picture of interface 2** | **Brief description of interface 2** |
|  | **Purpose**: User must fill this form to get access for monthly payment.  **Flow**: User data that required to fill this form should be matching.  **Good practices**:   * Without using complex or system-oriented words use words that are easy to read and understand. |
| **Picture of interface 3** | **Brief description of interface 3** |
|  | **Purpose**: User can provide his Payment detail from this form, and he can do his payments.  **Flow**: After customer choose to pay membership payment. He had to Fill this form and complete the payment.  **Good practices**:   * Use meaningful label names. Each word should start with a capital letter, and there should not be any spelling mistakes. |
| **Picture of interface 4** | **Brief description of interface 4** |
|  | **Purpose**: After customer getting late to return a book on time, he had to pay late payments when he is returning the book. User can provide his Payment detail from this form, and he can do his payments.  **Flow**: User must provide his user id, book id and password to system to identify the payment.  **Good practices**:   * Ensure that an interface has only one primary focus. Avoid doing multiple tasks from a single interface. |
|  | **Picture of the interface\_5** | **Brief description of interface\_5** |
|  | **Purpose**: User can provide his Payment detail from this form, and he can do his payments  **Flow**: After system identify the customer and book, he had to pay the amount that system is giving.  **Good practices**:   * Ensure that the users do not have to think or work too hard to use an interface. |
| **Picture of the interface\_6** | **Brief description of interface\_6** |
|  | **Purpose**: This invoice is the confirmation of the payment.  **Flow**: After doing successful payment, he will reserve this invoice and user can download it to his device.  **Good practices**:   * Have a meaningful title for the interface. |
| **Picture of the interface\_7** | **Brief description of interface\_7** |
| **Sprint 2** | **Purpose**: This is admin view of membership payment.  **Flow**: Only admin can access to this data, and he has to logging to the system as admin.  **Good practices**:   * Ensure that an interface has only one primary focus. Avoid doing multiple tasks from a single interface. |
| **Picture of the interface\_8** | **Brief description of interface\_8** |
|  | **Purpose**: This is admin view of late payment data.  **Flow**: admin can update the table from date vise, and he can detect the successful late payment.  **Good practices**:   * Keep the interfaces simple. An interface should not contain irrelevant information |
| **Registration No** | IT20150198 | |
| **Student**  **Name** | Mudalige.A.K.M.M.M | |
| **Function(s)**  **Note**: Include the functions required to complete for **sprint 1** & **sprint 2** | **Sprint 1**  Implementing all the features of customer side issue books.   * Book request home page * Book request form page * Available book page   **Sprint 2**  Implementing the all the features of admin side issue books.   * Book requests delete page * Admin book details search page * Admin update page * Report generate page | |
| **Sprint 1** | **Picture of interface 1** | **Brief description of interface 1** |
|  | **Purpose**: Customer can request for barrow book.  **Flow**: customer must choose book to barrow. After he can access to request book form.  **Good practices**:   * Keep the interfaces simple. An interface should not contain irrelevant information. |
| **Picture of interface 2** | **Brief description of interface 2** |
|  | **Purpose**: This is book request form for customer to barrow book.  **Flow**: First he must choose the book. he must input user id book id and wanted date. After he can request for book.  **Good practices**:   * Ensure that an interface has only one primary focus. Avoid doing multiple tasks from a single interface. |
| **Picture of interface 3** | **Brief description of interface 3** |
|  | **Purpose**: User can view the available book.  **Flow**: When user need to barrow books, he can choose the book from this page.  **Good practices**:   * Keep the interfaces simple. * Ensure that the users do not have to think or work too hard to use an interface. |
| **Sprint 2** | **Picture of the interface\_4** | **Brief description of interface\_4** |
|  | **Purpose**: Admin can update the availability of books.  **Flow**: Only admin can view data and he must log to the system as admin.  **Good practices**:   * Have a meaningful title for the interface. |
| **Picture of the interface\_5** | **Brief description of interface\_5** |
|  | **Purpose**: Admin can search the ID and he can choose the book details.  **Flow**: After choosing the relevant book by admin he can update the availability of book.  **Good practices**:   * Use meaningful label names. Each word should start with a capital letter, and there should not be any spelling mistakes. |
| **Picture of the interface\_6** | **Brief description of interface\_6** |
|  | **Purpose**: Admin can delete his request using the form.  **Flow**: If admin want to delete the request form.  **Good practices**:   * Ensure that an interface has only one primary focus |
| **Picture of the interface\_7** | **Brief description of interface\_7** |
|  | **Purpose**: This interface is use to generate daily report about return books.  **Flow**: Administrator can search books issue date and get relevant date his table after he can download report.    **Good practices**:   * multiple tasks from a single interface. |
| **Registration No** | IT20165970 | |
| **Student**  **Name** | **Wickramarathna.I.E** | |
| **Function(s)**  **Note**: Include the functions required to complete for **sprint 1** & **sprint 2** | **Sprint 1**  Implementing all the features of manage books.   * Add new books * Update book details * Search book details   **Sprint 2**   * Manage e-books * Manage authors * Report generation part | |
| **Sprint 1** | **Picture of interface 1** | **Brief description of interface 1** |
| Graphical user interface  Description automatically generated | **Purpose**: This interface will be used to add new book to the system.  **Flow**: Initially, the administrator  needs to include the subject of the book, the ISBN number, the name of the author, the number of copies, the volume of the book and the copyright of the book. After entering the required values an administrator will need to save the entered data. All the entered data is represented below in tabular form.  **Good practices**:   * The interface was designed by focusing on one specific task, which is the addition of the new books to the library. * The labels were given meaningful names, first letter was capitalized. In addition, it was made sure that there are no spelling mistakes. * Given the option to select the relevant book category and number of copies from the drop-down list, without having to type in all the values. |
| **Picture of interface 2** | **Brief description of interface 2** |
| Graphical user interface  Description automatically generated | **Purpose**: Managing search book operation are in the library.  **Flow**: The table displayed contains of all books. admin can search the book details of using ISBN Number. And the admin is allowed to select the record and update it.  **Good practices**:   * The interface was designed by focusing on one specific task, which is the search for books in the library. * This interface has a table filtering option. * Simple interface. |
| **Picture of interface 3** | **Brief description of interface 3** |
| Graphical user interface  Description automatically generated | **Purpose**: Managing the update book details.  **Flow**: The admin must select the necessary record. once record selected, the data will be loaded into the fields another page. If the admin wants to update the data, they can edit the number of copies of the book and click on the “UPDATE” button. Then update details display the down below.  **Good practices**:   * Given the option to select the relevant book category and number of copies from the drop-down list, without having to type in all the values. * The interface was designed by focusing on one specific task, which is the update of the number of book copies to the library. * This interface has a meaningful title. |
| **Picture of interface 4** | **Brief description of interface 4** |
| Graphical user interface, application  Description automatically generated | **Purpose**: The administrator can see the books that members recommended for the library.  **Flow**: Once the member has recommended the book, it will be displayed on this page. Admin can also search for members' recommendations by date and delete the recommendations.  **Good practices**:   * The interface was designed by focusing on one specific task, which is the view the recommended books to the library. * This interface has a table filtering option by date. * When admin click on the Delete button, specific row going to be removed in DB. |
| **Sprint 2** | **Picture of the interface\_5** | **Brief description of interface\_5** |
| Graphical user interface, application  Description automatically generated | **Purpose**: This page is an e-book page. Members of this library, they have can ability to read e-books online using this page.  **Flow**: This interface will be used to manage e-books in the system. The administrator can add the e-books and delete e-books monthly.  **Good practices**:   * The interface was designed by focusing on one specific task, which is the add e-books to the library. * Simple interface. * When admin click on the Delete button, specific e-book going to be removed in DB. |
| **Picture of the interface\_6** | **Brief description of interface\_6** |
| Graphical user interface  Description automatically generated | **Purpose**: This interface will be used to add new authors to the system.  **Flow**: Initially, the administrator needs to include the author’s name, category, book title, edition, the number of copies, the number of pages, year of publication and the name of the publisher. After entering the required values an administrator will need to save the entered data using “Insert the information” button.  **Good practices**:   * The interface was designed by focusing on one specific task, which is the addition of the new authors to the library. * The labels were given meaningful names, first letter was capitalized. In addition, it was made sure that there are no spelling mistakes. * Given the option to select the relevant book category, year of publication and number of copies from the drop-down list, without having to type in all the values. * Simple interface. |
| **Picture of the interface\_7** | **Brief description of interface\_7** |
| Graphical user interface  Description automatically generated | **Purpose**: Display all author details are in the library.  **Flow**: The table displayed contains of all authors. admin can search the author details of using by author’s name.  **Good practices**:   * The interface was designed by focusing on one specific task, which is the display all author details in the library. * This interface has a table filtering option. * Simple interface. |
| **Picture of the interface\_8** | **Brief description of interface\_8** |
| Graphical user interface, text, application, email  Description automatically generated | **Purpose**: This interface will be used to display the book list of the library.  **Flow**: Admin can generate the book stock report using this interface.  **Good practices**:   * The interface was designed by focusing on one specific task, which is the report generate of the stock of books. * Simple interface. |
|  | **Picture of interface 9** | **Brief description of interface 9** |
| Graphical user interface, text, application  Description automatically generated | **Purpose**: This interface will be used to display the authors list of the library.  **Flow**: Admin can generate all the author details report using this interface.  **Good practices**:   * The interface was designed by focusing on one specific task, which is the report generate of the author details. * Simple interface. |
| **Registration No** | IT20173036 | |
| **Student**  **Name** | Peiris.H.T.D.P | |
| **Function(s)**  **Note**: Include the functions required to complete for **sprint 1** & **sprint 2** | **Sprint 1**   * Home page * View profile * Update member details   **Sprint 2**   * Recommendation books * Borrowed books * Returned books * Report generation part | |
| **Sprint 1** | **Picture of interface 1** | **Brief description of interface 1** |
| Graphical user interface, diagram, application, PowerPoint  Description automatically generated | **Purpose**: This interface will be used to access the several task in home page.  **Flow**: The user can go to the home page and click on the several tasks as he/she wishes. Such as borrow books, e books, return books, search books and payments. As well as user can logout to the system here.  **Good practices**:   * Use meaningful label names. Each word should start with a capital letter. * Ensure that the users do not have to think or work too hard to use an interface. * Have a meaningful task in the interface. |
| **Picture of interface 2** | **Brief description of interface 2** |
| Graphical user interface, text, application, email  Description automatically generated | **Purpose**: This interface will be used to show the membership details.  **Flow**: here the member can see the information entered the system and if they want to change the any information, they can click on the update icon.  **Good practices**:   * Keep the interfaces simple. * Ensure that the users do not have to think or work too hard to use an interface. |
| **Picture of interface 3** | **Brief description of interface 3** |
| **Graphical user interface  Description automatically generated** | **Purpose**: This interface will be used to update the membership details into the system.  **Flow**: The user can update to the Reg no, Reg Date, Name, Email, Phone, Area of Interest, Address, NIC and shift. The user can click on the update button, then the data will be updated.  **Good practices**:   * The interface was designed by focusing on one specific task, which is the update the member details. * The labels were given meaningful names, and the first letter of each word was capitalized. * Have a meaningful title for the interface. |
| **Sprint 2** | **Picture of the interface\_4** | **Brief description of interface\_4** |
| Graphical user interface, application  Description automatically generated | **Purpose**: This interface will be used to recommend the book for library. As well as delete the recommendation.  **Flow**: the member is required to enter the Author’s name, Book name, Select the subject, edition, publications, and year. The member can click the recommend button. If you think that recommendation is unnecessary, you can delete it from there.  **Good practices**:   * Have a meaningful title for the interface. * Ensure that an interface has only one primary focus. Avoid doing multiple tasks from a single interface. |
| **Picture of the interface\_5** | **Brief description of interface\_5** |
| **Graphical user interface, table  Description automatically generated** | **Purpose**: This interface will be used to search the data of the books previously used by the member.  **Flow**: If the user wants, he/she can search by the date to see the books he/she has borrowed or the date on which those books should be returned.  **Good practices**:   * The interface was designed by focusing on one specific task, which is the search the borrowing history. * Each word should start with a capital letter, and there should not be any spelling mistakes. * Keep the interfaces simple. |
| **Picture of the interface\_6** | **Brief description of interface\_6** |
| Table  Description automatically generated | **Purpose**: This interface will be used to print as a report a list of books borrowed by the user.  **Flow**: If the user wants to see the list of borrowed books, then he/ she can click on the “Print” button.  **Good practices**:   * Keep the interfaces simple. * The interface was designed by focusing on one specific task. |