|  |  |  |
| --- | --- | --- |
| **Group No:** | **REG/WE/27** | |
| **Brief description of the project** | **This project is a web application for the LORENZO RENT-A-CAR Company. Earlier in this vehicle rental center they had a manual system which the record keeping was done using books and excel sheets. As they are now enhanced their work quality and the endorsements of the service center, they suggested our team of developers to build up on a fully automated system for the internal and external operations of the Car rental system. Through this system each function will be assigned to a specific group of individuals and this system ensures that the integrity of the system and also the security is very high through admin-based logins and backups. This is a fully functional system aiming even the smallest problem that can occur in-house and it provides ease to the users because it is user friendly. There will be parts including Booking management, Customer management, Vehicle Fleet management, Employee management, and Vehicle return management. All the database records are maintained used a highly secure based procedure. In this system daily, monthly, and yearly records could be generated in each function. It helps in venturing through statistics clearly in any of the function. This system mostly tends to generate automatic calculations and results which ensures that the human-based errors will be at a bay when using. This system was built up using Java as the language and MySQL as the database type, in order to make this product more efficient.** | |
| **INDIVIDUAL DETAILS OF GROUP MEMBERS** | | |
| **Registration No** | **IT20140366** | |
| **Student Name** | **Tharushika Devindi M.K.S.** | |
| **Function(s)**  **Note**: Include the functions required to complete for **sprint 1** & **sprint 2** | **Sprint 1**   * Customer Registration / Add Customer * Customer Login * Customer Profile * Customer List & Delete Customer * Update Customer Details * Generate Customer Report   **Sprint 2**   * Descriptions of Services & Add Service Request * Service Request Profile * Service Request List & Delete Service Request * Update Service Request Details * Generate Service Request Report | |
| **Sprint 1** | **Picture of interface 1** | **Brief description of interface 1** |
| Graphical user interface  Description automatically generated | **Purpose**: This Interface will used to add new customer to the system.  **Flow**: Initially the customer / admin required to fill the customer Full name, Email, Contact number, Register date, Username and Password. Then the customer / admin needs to check the “I agree” checkbox. After entering the values and check the checkbox, the customer / admin needs to click “REGISTER” button to register as new customer. And the Username and Password will be given to the new customer. (There will be validations for every input field)  **Good practices**:   * The interface was designed to one task, which is adding new customer to the system. * The interface is very simply designed. This interface does not contain irrelevant information. * Expresses error messages in an easy-to-understand way. Clearly notifies the error with hints to fix the problem. |
| **Picture of interface 2** | **Brief description of interface 2** |
|  | **Purpose**: This interface will be used to login to the customer site of the web application.  **Flow**: In hear the customer is required to enter the customer username and password. After entering the correct details, the customer needs to click on the “Log In” button. Then system will redirect to the customer profile page.  **Good practices**:   * The interface was designed by focusing on one task, which is the customer login. * The interface was given meaningful title. * The interface is very simply designed. This interface does not contain irrelevant information. |
| **Picture of interface 3** | **Brief description of interface 3** |
| **Graphical user interface  Description automatically generated** | **Purpose**: This interface will be used to view the customer profile.  **Flow**: After the successful login, the customer will redirect to this page and he/she can view his/her details and, he/she can update or delete his/her profile using “UPDATE” or “DELETE” buttons.  **Good practices**:   * The labels were given meaningful names and the first letter of each word is capitalized. In addition, it was made sure that there are no spelling mistakes. * The interface was given meaningful title. * The customer does not have to think or work too hard to use this interface. |
| **Picture of interface 4** | **Brief description of interface 4** |
| Graphical user interface, application, website  Description automatically generated | **Purpose**: This interface will be used to view all the customer details that registered to the system.  **Flow**: When the system admin clicks on the “Customer Management” button, then the system will redirect to this page.  In here we can see all the customers that registered to the system. Admin can simply search about the customer details using the search bar and, he / she can update or delete customer information by clicking on the clickable icons update or delete. When clicks on the update icon, then it will redirect to “Update Customer Details” page. When clicks on the delete icon, then popup a message as “Are you sure you want to delete?”. When the admin clicks on the “Add Customer” button, then the system will redirect to the customer registration page. In this time system admin also can insert customers to the system by using customer registration page. The system admin can generate a monthly report by clicking “Generate Customer Report” button.  **Good practices**:   * The components (buttons, labels, etc.) used in the interface are large enough to interact, ensuring that all capabilities are accessible to users, including those with low vision. * The interface was given meaningful title. * Words that are easy to read and understand are used without using complex or system-oriented words. |
| **Picture of interface 5** | **Brief description of interface 5** |
| **Graphical user interface  Description automatically generated** | **Purpose**: This interface will be used to update the customer details.  **Flow**: After clicking the update icon on the customer list page, system will redirect to this page. Admin can select the data fields and change the values. After clicking the 'Update' button, "Do you want to save the changes?" A message popup as. The updated data saved to the database by click on the “Save” button.  **Good practices**:   * The interface was designed by focusing on one task, which is the update customer details. * The interface is very simply designed. This interface does not contain irrelevant information. * The customer does not have to think or work too hard to use this interface. |
| **Picture of interface 6** | **Brief description of interface 6** |
| **Graphical user interface  Description automatically generated** | **Purpose**: This interface will be used to view the report of all the customers.  **Flow**: After clicking the “Generate Customer Report” button on the customer list page, system will redirect to this page. Customer details which are stored and displayed is required for future use. Therefor system admin generates a monthly customer report.  **Good practices**:   * The interface was given meaningful title. * The interface was designed by focusing on one task, which is the report of all the customers. * This interface does not contain irrelevant information. |
| **Sprint 2** | **Picture of the interface\_7** | **Brief description of interface\_7** |
| Graphical user interface  Description automatically generated | **Purpose**: This interface will be used to view the descriptions of services and to view the service request form.  **Flow**: When the customer clicks on the “Services”, then the system will redirect to this page. In here we can see all the services and descriptions of them. Furthermore, we can see service request form. Customer can fill this form and submit it by clicking “SUBMIT” button. If the customer needs to clear the previous data of the form, he/she can click the “RESET” button.  **Good practices**:   * Allows them to select values whenever possible, without asking the user to type values. * The customer does not have to think or work too hard to use this interface. * Without asking the users to type in values, they were given the option to choose to relevant service from the drop-down list. |
| **Picture of the interface\_8** | **Brief description of interface\_8** |
| **Graphical user interface  Description automatically generated** | **Purpose**: This interface will be used to view the service request profile.  **Flow**: After submitted the service request form, customer will redirect to this page and he/she can view his/her service request details and, he/she can update or delete his/her service request profile using “UPDATE” or “DELETE” buttons.  **Good practices**:   * The labels were given meaningful names and the first letter of each word is capitalized. In addition, it was made sure that there are no spelling mistakes. * The interface was given meaningful title. * The customer does not have to think or work too hard to use this interface. |
| **Picture of the interface\_9** | **Brief description of interface\_9** |
| **Graphical user interface  Description automatically generated** | **Purpose**: This interface will be used to view all the service request details that submitted to the system.  **Flow**: When the system admin clicks on the “Service Request” button, then the system will redirect to this page.  In here we can see all the service requests that submitted to the system. Admin can simply search about the requested services details using the search bar and, he / she can update or delete service request information by clicking on the clickable icons update or delete. When clicks on the update icon, then it will redirect to “Update Service Request Details” page. When clicks on the delete icon, then popup a message as “Are you sure you want to delete?”. When the admin clicks on the “Add Service Request” button, then the system will redirect to the services page. In this time system admin also can insert service request to the system by using services page. The system admin can generate a monthly service request report by clicking “Generate Report” button.  **Good practices**:   * The interface was given meaningful title. * Words that are easy to read and understand are used without using complex or system-oriented words. * The components (buttons, labels, etc.) used in the interface are large enough to interact, ensuring that all capabilities are accessible to users, including those with low vision. |
| **Picture of the interface\_10** | **Brief description of interface\_10** |
| Graphical user interface, website  Description automatically generated | **Purpose**: This interface will be used to update the service request details.  **Flow**: After clicking the update icon on the service request list page, system will redirect to this page. Admin can select the data fields and change the values. After clicking the “UPDATE” button, "Do you want to save the changes?" A message popup as. If he/she want to cancel the updating process he/she can simply click “CANCEL” button.  **Good practices**:   * The components (buttons, labels, etc.) used in the interface are large enough to interact, ensuring that all capabilities are accessible to users, including those with low vision. * The interface was designed by focusing on one task, which is the update the service request details. * The labels were given meaningful names. In addition, it was made sure that there are no spelling mistakes. |
| **Picture of the interface\_11** | **Brief description of interface\_11** |
| **Graphical user interface  Description automatically generated** | **Purpose**: This interface will be used to view the report of the service request.  **Flow**: After clicking the “Generate Report” button on the service request profile page, system will redirect to this page. Service request details which are stored and displayed is required for future use. Therefor system admin generates a monthly service request report.  **Good practices**:   * The interface was given meaningful title. * The interface was designed by focusing on one task, which is the report of all the customers. * This interface does not contain irrelevant information. |
| **Registration No** | **IT20178154** | |
| **Student Name** | **Dilshan P.A.D.S.D** | |
| **Function(s)**  **Note**: Include the functions required to complete for **sprint 1** & **sprint 2** | **Sprint 1**   * Implement all features of Admin management   **Sprint 2**   * Admin delete/admin list * Add Booking * Booking List * Update Booking | |
| **Sprint 1** | **Picture of interface 1** | **Brief description of interface 1** |
| Graphical user interface, website  Description automatically generated | **Purpose**: This interface will be used to login to the administrative site of the web application  **Flow**: initially, admin of the Lorenzo Rent-A-Car website need to enter the admin username and password. After entering the correct details admin need to click the Login Button. then system will redirect to admin Profile page  **Good practices**:   * The interface was designed by focusing on one task, which is admin Login |
| **Picture of interface 2** | **Brief description of interface 2** |
| Graphical user interface, application, website  Description automatically generated | **Purpose**: This Interface will used to add new system admins to the system.  **Flow**: In hear the senior system admin required to fill the Full Name, Email, Contact Number, Admin User Name, Password. After entering the values admin needs to click ‘REGISTER’ button to register the new system admin. And the User name and Password will be given to the new admin. (There will be Validations for every input field)  **Good practices**:   * The interface was designed to one task which is adding new system admins to the system * The labels were given meaningful names and the first letter of each word is capitalized. |
| **Picture of interface 3** | **Brief description of interface 3** |
| Graphical user interface, website  Description automatically generated | **Purpose**: This interface will be used to view the admin profile  **Flow**: after the successful login admin will redirect to this page and he/she can view his details and also he can update and delete his profile using Update Details button and Delete button  **Good practices**:   * The user details will automatically added to the hint text areas * The labels were given meaningful names * Without using complex words there is simple words that can easily understand in the labels |
| **Picture of interface 4** | **Brief description of interface 4** |
| **Graphical user interface, website  Description automatically generated** | **Purpose**: This interface will be used to update the admin details  **Flow**: after clicking the ‘Update Details’ button system will redirect to this page . admin can select the data fields and change the values. After that admin can update the data in the database by clicking ‘Update’ Button. If he want to cancel the updating process he can simply click ‘Cancel’ button.  **Good practices**:   * Interface was designed by focusing on one specific task and its admin details update * The labels were given meaningful names and first letter of each word was capitalized |
| **Sprint 2** | **Picture of the interface\_5** | **Brief description of interface\_5** |
| Graphical user interface  Description automatically generated | **Purpose**: This interface will be used to Delete the admin details  **Flow**: after clicking the ‘Delete’ button system will redirect to this Page. admin can delete the profile by clicking ‘Delete’ button. If he wants to cancel the Deleting process he can simply click ‘Cancel’ button  **Good practices**:  Admin do not have to think or work too hard to use interface |
| **Picture of the interface\_6** | **Brief description of interface\_6** |
| Table  Description automatically generated | **Purpose**: This interface will be used to view the full list of system admins and we also can update and delete their records.  **Flow**: when the user click the system admin Management button system will redirect to this page. We can see the full list of system admins who registered in the system. Admin can simply search about the users using the search bar. And admin can Generate a full report by clicking ‘Generate Report’ button.  **Good practices**:   * User can search any detail in the database using search bar |
| **Picture of the interface\_7** | **Brief description of interface\_7** |
| Graphical user interface, website  Description automatically generated | **Purpose**: when a customer try to book a vehicle using a telephone call or by a email, system admin can easily book a vehicle easily by using this interface.  **Flow**: initially the system admin is required to fill the customer name , email , contact number, pickup location and Destination after that admin must choose service type and the vehicle type. After that he need to choose pickup date from the calendar and also pick up time. Then he need to pickup return date and drop off time. Message part is optional and there will be validation for every field. After all admin can click Submit Inquiry button  **Good practices**:   * The labels were given meaning full names * Admin can tick the service type and vehicle type without typing * Admin can select pickup date and return date easily from calendar. |
| **Picture of the interface\_8** | **Brief description of interface\_8** |
| **Table  Description automatically generated** | **Purpose**: This interface will be used to view the full list of Bookings and we also can update and delete their records.  **Flow**: when the admin click the Booking Management button system will redirect to this page. We can see the full list of bookings in the system. Admin can simply search about the bookings using the search bar and also he can update and delete bookings using update and delete buttons. admin can Generate a full report by clicking ‘Generate Report’ button    **Good practices**:   * User can search any detail in the database using search bar |
|  | **Graphical user interface  Description automatically generated** | **Purpose**: This interface will be used to update booking details  **Flow**: initially the system admin can update the customer name , email , contact number, pickup location and Destination after service type and the vehicle type. After that he need to choose pickup date from the calendar and also pick up time. Then he need to pickup return date and drop off time. Message part is optional and there will be validation for every field. After all admin can click Update Inquiry button   * **Good practices**: * The labels were given meaning full names * Admin can tick the service type and vehicle type without typing * Admin can select pickup date and return date easily from calendar |

|  |  |  |
| --- | --- | --- |
| **Registration No** | **IT20139858** | |
| **Student Name** | **Gunasinghe P.S.L** | |
| **Function(s)**  **Note**: Include the functions required to complete for **sprint 1** & **sprint 2** | **Sprint 1**   * Implement all features of vehicle fleet management   **Sprint 2**   * Add customer booking * Update booking * Remove booking * Customer profile | |
| **Sprint 1** | **Picture of interface 1** | **Brief description of interface 1** |
|  | **Purpose** : This interface used to add a new vehicle to the system by administrator  **Flow**: In this part administrator want to add a new vehicle to the system. before adding the vehicle, administrator want to login to the system using admin login. Then he wants to add vehicle brand, model, registration number, transmission, fuel type and picture of the vehicle. In here the massage is about brief note about the options of the vehicle  **Good practices**: administrator can easily add a vehicle to the system |
|  | **Picture of interface 2** | **Brief description of interface 2** |
| Table  Description automatically generated | **Purpose** : This interface will be used to vehicle fleet list  **Flow**: When user click add vehicle button in the interface 1, then it redirects to this page. In this vehicles list has all the vehicles by the system admin previously added. Admin can simply search any vehicle using search bar. Then, admin can generate a report using “Generate report” button. Also, admin can update and delete vehicle details.  **Good practices**: User can find a vehicle easily |
| **Picture of interface 3** | **Brief description of interface 3** |
| Graphical user interface, application  Description automatically generated | **Purpose :** This interface will be used to update vehicle details.  **Flow:** After clicking update button in vehicle list, then redirect to this page. The system administrator can update the vehicle. After he fill the form, mistakenly added wrong type of data he can change it in to correct. After he change it correct, he can save it using click in the ”Update Vehicle Details”.  **Good practices:**  Administrator can easily change the information. |
| **Picture of interface 4** | **Brief description of interface 4** |
| **Graphical user interface, application  Description automatically generated** | **Purpose** : This interface will be used to Delete the admin details  **Flow**: Click by the delete button in vehicle fleet list, then redirect to this page. In here admin want to remove a vehicle in the system then he can remove it click “Yes” in pop-up massage.  **Good practices** : If user mistakenly going to delete wrong vehicle, then the user can select “No” button in pop-up warning massage. |
| **Picture of interface 5** | **Brief description of interface 5** |
|  | **Purpose** : This interface is used to when the customer book a vehicle by himself  **Flow**: Customer must add his/her name, email, contact number, pickup location ,destination, service type and vehicle type. Then customer need to add his/her pickup date, pickup time , return date and return time. The massage is optional. Every field has validation. After filling the form, customer can submit the form, but he can only see his/her booking details only.  **Good practices**: Customer cannot access the other booking details. This is the main different between admin booking and customer booking |
| **Picture of interface 6** | **Brief description of interface 6** |
|  | **Purpose** : This interface is used to update and delete booking details  **Flow**: when the user click submit inquiry button in interface 5, then it redirects to this interface. If customer want to update his/her data, he/she can easily update it. Also, customer can delete their own booking and cancel it.  **Good practices** : customer can easily update or delete his own data. |

|  |  |  |
| --- | --- | --- |
|  | **Picture of interface 7** | **Brief description of interface 7** |
|  | **Purpose** : After booking, customer received a booking successful massage  **Flow**: After a successful booking system will redirect to this page an it will show "Your booking was added successfully" message. And customer can go back to previous page by clicking Go back button. He can go to home page by clicking the Return Home Page button  **Good practices**: customer can easily identify the booking was added or not. |
|
|

|  |  |  |
| --- | --- | --- |
| **Registration No** | IT20139544 | |
| **Student Name** | Kapukotuwa.S.A.A.H | |
| **Function(s)**  **Note**: Include the functions required to complete for **sprint 1** & **sprint 2** | **Sprint 1**   * Vehicle return details * Update or delete the details of returned vehicles * Generate report of returned vehicles   **Sprint 2**   * Feedback * Update feedback * Thanking Page | |
| **Sprint 1** | **Picture of interface 1** | **Brief description of interface 1** |
| Capture 3 | **Purpose**: This interface will be used to add the details about returned vehicles.  **Flow**: This job is assigned to admin. After a customer returned a vehicle admin is required to fill the customer name,email address, contact number,pickup and return date, return time and also the vehicle and service type. After then admin can enter the details by click on submit button.  **Good practices**:   * Admin can easily manage the vehicles. |
| **Picture of interface 2** | **Brief description of interface 2** |

|  |  |  |
| --- | --- | --- |
|  | **Picture of interface 2** | **Brief description of interface 2** |
| Capture 4 | **Purpose**: This interface will be used to update and delete the details about returned vehicles which were added.  **Flow**:After the admin click on submit button the interface 1, then it redirect to this interface. If admin need to update or delete the details about returned vehicle, it can be done in here.  **Good practices**:   * Admin can update and delete the details of returned vehicles and manage properly. |
| **Picture of interface 3** | **Brief description of interface** |

|  |  |  |
| --- | --- | --- |
|  | **Picture of interface 3** | **Brief description of interface 3** |
| Capture 5 | **Purpose**: This interface will be used to veiw the list of returned vehicle details.  **Flow**: After admin added the details of returned vehicle , the details are listed in here. In a case admin can delete or update the details of this list. And also admin can generate a report of returned vehicles.  **Good practices**:   * Admin can easily update and delete the details. * Admin can generate a report of returned vehicle details. |
|  |  |

|  |  |  |
| --- | --- | --- |
| **Sprint 2** | **Picture of interface 4** | **Brief description of interface 4** |
| Capture | **Purpose**: This interface will be used to add feedback by customer.  **Flow**: A customer can add their name, experience as a message and ratings as their feedback in this interface. After all the customer can submit their feedback. In this interface there is an overall user rating and also feedback and ratings which were added by other customers.  **Good practices**:   * Customer can get an idea of this system by previously added feedback and ratings. * Customer can rate the service and add their feedback. |
| **Picture of interface 5** | **Brief description of interface 5** |

|  |  |  |
| --- | --- | --- |
|  | **Picture of interface 5** | **Brief description of interface 5** |
| Capture 6 | **Purpose**: This interface will be used to view the feedback which was added.  **Flow**: After the customer submit her/his feedback in interface 4, it redirects to this interface. In here she/ he receive a thanking message. If the customer need to view their feedback, they can click on view button and view their feedback.  **Good practices**:   * Customer can view the feedback which was previously added by him/her. |
| **Picture of interface 6** | **Brief description of interface 6** |

|  |  |  |
| --- | --- | --- |
|  | **Picture of interface 6** | **Brief description of interface 6** |
| Capture1 | **Purpose**: This interface will be used to update or delete the feedback which was added previously by the customer.  **Flow**: After the customer view his/her own feedback and he/ she need to update or delete feedback, it can be done in this interface.  **Good practices**:   * Customer can easily update or delete their his/her own feedback which was added previously. |
|  |  |