**ABSTRACT**

The ever increasing vehicle population gives rise to the number of traffic problems. As the count of multiplexes and malls is rising and making the cities crowded, the traffic problems faced by the people are countless, mostly encountered one of them being the parking space availability. There is never a certainty of getting a place to park the vehicles by owners when they get to the destination. The current parking facility is unable to cope up with the influx of vehicles on road.

Presently, India doesn’t have the Smart Parking system with wireless sensor networks for full automation. Due to ever increasing influx of vehicles, there is a dire need of smart parking system management tools to manage the heavy ingress of vehicles. There are various other smart parking system alternatives like RFID enabled parking system, E-parking system, fuzzy logic implementation in parking system etc. which are, due to some technical backwardness, not feasible in India presently.

One of the solution to the aforesaid problem is Multistorey car park buildings near big landmarks where there is always a rush, like malls, multiplexes etc, in order to take the load off the roads. But the glitch here is that not many people are aware of and have faith in this facility which is recently introduced in India.

To make parking comfortable and bring these car parks to the fore, the project team has come up with ParkMeRight. A project that is meant for user friendly management of these car park systems along with letting the vehicle owners be in touch with these systems 24×7 by providing slot availability if pre-booked. Hence, the users would be able to park their cars comfortably without any hassle.

**LIST OF FIGURES**

|  |  |  |  |
| --- | --- | --- | --- |
| **S. No.** | **Figure No.** | **Name of Figure** | **Page No.** |
| 1. | 1.1 | Exterior Car Park | 3 |
| 2. | 1.2 | Interior Car Park | 3 |
| 3. | 1.3 | Vehicle Lifts | 4 |
| 4. | 2.1 | Disc Shaped Parking System | 10 |
| 5. | 2.2 | Example of Parking System Reservation Message | 14 |
| 6. | 3.1 | Position of JSP container and JSP files | 16 |
| 7. | 3.2 | JSP Processing | 17 |
| 8. | 3.3 | Android Environment | 19 |
| 9. | 3.4 | Android Architecture | 19 |
| 10. | 3.5 | Use Case Diagram | 23 |
| 11. | 5.1 | Database Structure | 29 |
| 12. | 6.1 | Setting JDK runtime environment for the web application | 33 |
| 13. | 6.2 | Setting Server runtime environments for the web application project | 34 |
| 14. | 6.3 | Selecting Tomcat Server As the New Server Runtime | 35 |
| 15. | 6.4 | J2EE Project under configured server runtime environment | 36 |
| 16. | 6.5 | Creating new dynamic web application in new server runtime environment | 37 |
| 17. | 6.6 | Tree Hierarchy of Project | 39 |
| 18. | 6.7 | Flow Chart | 40 |
| 19. | 6.8 | Android Project Execution | 47 |
| 20. | 7.1 | RESTEasy Client | 48 |
| 21. | 7.2 | Login Web Service | 49 |
| 22. | 7.3 | Get Details Web Service | 50 |
| 23. | 7.4 | Request to fetch floor plan web service | 51 |
| 24. | 7.5 | Response of the fetch floor plan web service | 51 |
| 25. | 7.6 | Request to the forgot password web service | 52 |
| 26. | 7.7 | Response from the forgot password web service | 52 |
| 27. | 7.8 | Request for submitting user registration data | 53 |
| 28. | 7.9 | Response by the registration web service | 53 |
| 29. | 7.10 | Login Page | 54 |
| 30. | 7.11 | Sign Up Page | 55 |
| 31. | 7.12 | Terms and Conditions | 56 |
| 32. | 7.13 | Main Page | 57 |
| 33. | 7.14 | Booking a slot | 58 |
| 34. | 7.15 | Profile Page | 59 |
| 35. | 7.16 | Forgot Password Facility | 60 |
| 36. | 7.17 | Android App Login | 61 |
| 37. | 7.18 | Registration Page | 62 |
| 38. | 7.19 | Advance Booking of Slot | 63 |
| 39. | 7.20 | Forgot Password | 64 |

**LIST OF TABLES**

|  |  |  |  |
| --- | --- | --- | --- |
| **S. No.** | **Table Number** | **Table Title** | **Page Number** |
| 1. | 4.1 | Software Requirements | 14 |
| 2. | 4.2 | Hardware Requirements | 14 |