**Online Ticketing System**

**For Issue Resolving**

Digital Internship in **Exavalu**

**GROUP - F**

Team members:

**AMIT KUMAR SHARMA**

**AQUIB JAWED**

**PRASUN GHOSH**

**SAYANI ROY**

**SNEHA KUMARI**

**INTRODUCTION**

A ticketing system is a software program that a customer support team uses to create, manage, and maintain a list (or lists) of customer problems.

A ticketing system provides facility to lodge an issue that is lodged by a CSR. The system tracks the status of each ticket as support staff members work on solving the issue. The system records all the information from lodging a ticket till the resolving of it and user providing feedback regarding the service.

When customer support agents’ close tickets, they don’t just disappear. Tickets contain valuable data that can give you insight into your customers and company.

**OBJECTIVE**

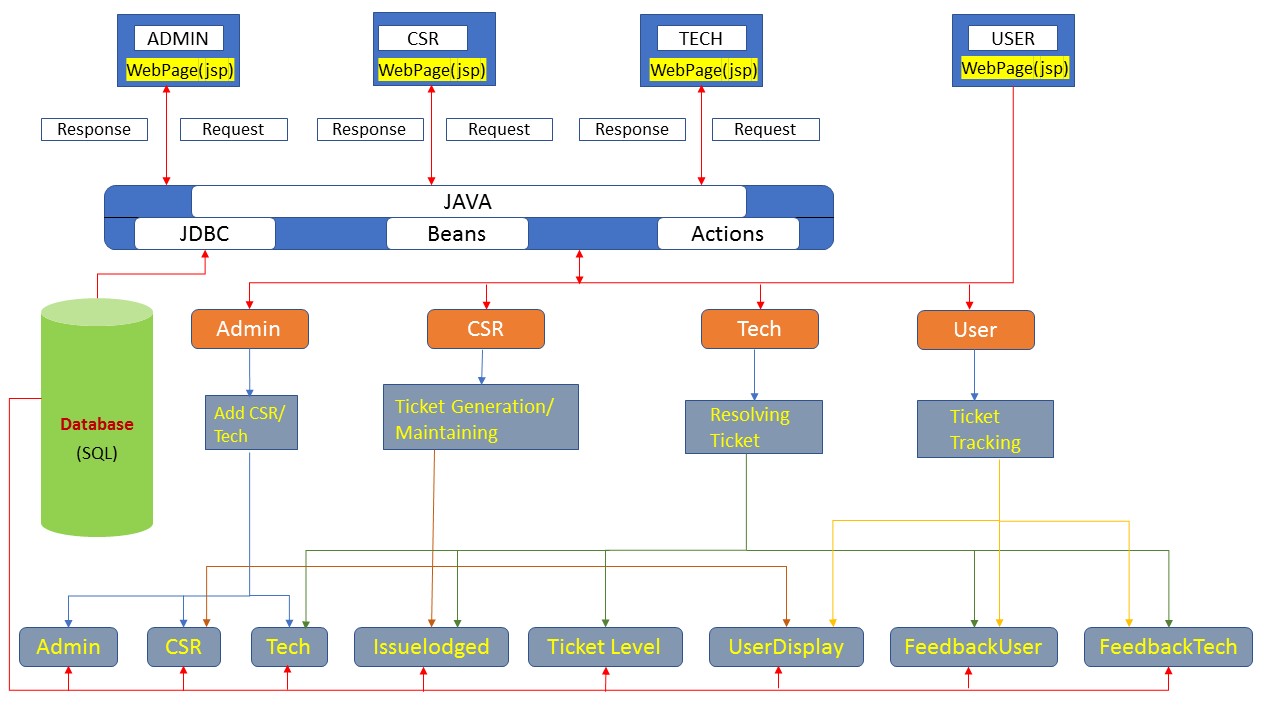
The objective of the project is to automate the process of issue resolving with minimal effort by providing easy to use interface in simplistic steps and spontaneous response generation with a provision of providing feedback.

**BENEFITS**

1. **Centralized System:**  
   As the ticketing system was organized centrally there is no need of hassles through spreadsheets or email inboxes trying to figure out the issues and their resolution processes.
2. **Automatic management of tasks:**  
   Using the ticketing system to manage the processes like ticket categorization and prioritization, ticket routing, alerts and notifications, and ticket status management will support the agents to spend more time resolving the actual support issues.
3. **Better Customer Experience:**  
   It will assist in improving requestors' perceptions for the support experience by troubleshooting the basic steps of agent-assisted support in terms of consistency and transparency along with information.
4. **Drives Staff Accountability and Training:**  
   The ticketing system provides managers with a view into the work of the employees which can help brighten the staff development opportunities.
5. **In-Built tracking and reporting:**  
   The data created by the software helps to drive the resourcing decisions, process improvement and enable the tuning of support processes to increase customer satisfaction.

**FEATURES**

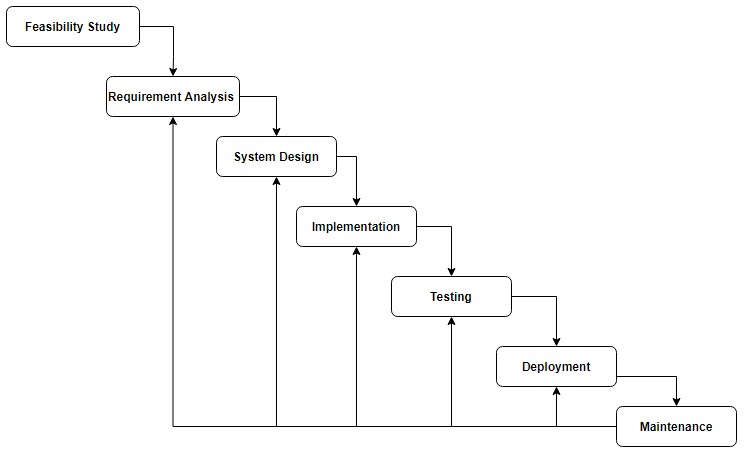
* The system provides the tentative date under which the issue will be resolved as soon as the issue is lodged and ticket is raised.
* Since CSR reports the problem, customer/client effort is cut off, they can check the status of the service by the issue Id that they have been provided and also provide feedback for the same.
* Online ticketing system provides provision for ticket escalation that helps in proper addressal of the issue.
* CSR and Technician logging into the system are provided access by Admin, therefore only identified personnel have the access.
* The system provides transparency by allowing technicians to provide feedback related to the issue that they worked on which will also be visible to the user.

**SYSTEM ARCHITECTURE**

**Software Model**

There are different phases of this Software model and they are as follow:

1. **Requirement gathering & analysis:** In this phase, we gathered the requirements, according to the project need.
2. **Design:** In the design phase, we designed the system using diagrams like Data Flow diagram, E-R diagram, etc.
3. **Implementation:** The requirements gathered were implemented by coding.
4. **Testing:** After completing the coding phase, software testing was done. We used Junit for testing.
5. **Deployment:** After completing all the phases, software is deployed to its work environment.
6. **Review:** In this phase, after the product deployment, we started review phase in which we check the behaviour and validity of the developed product. And if there are any error found then the process starts again from the requirement gathering.
7. **Maintenance:** In the maintenance phase, after deployment of the software in the working environment there may be some bugs, some errors or new updates are required.



Software Life Cycle Model

# **THEORETICAL BACKGROUND**

## The Project is comprised of the following:

1. **HTML**: The Hypertext Mark-up Language, or HTMLis the standard mark-up language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.
2. **CSS**: Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a mark-up language such as HTML.CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.
3. **JSP:** JSP technology is used to create web application just like Servlet technology. It can be thought of as an extension to Servlet because it provides more functionality than servlet such as expression language, JSTL, etc.
4. **JS:** JavaScript is a programming language that is one of the core technologies of the World Wide Web, alongside HTML and CSS. Over 97% of websites use JavaScript on the client side for web page behaviour, often incorporating third-party libraries. All major web browsers have a dedicated JavaScript engine to execute the code on users' devices.
5. **jQuery:** jQuery is a fast, small, and feature-rich JavaScript library. It makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers. With a combination of versatility and extensibility, jQuery has changed the way that millions of people write JavaScript.
6. **AJAX:**is a set of web development techniques that uses various web technologies on the client-side to create asynchronous web applications. With Ajax, web applications can send and retrieve data from a server asynchronously (in the background) without interfering with the display and behaviour of the existing page. By decoupling the data interchange layer from the presentation layer, Ajax allows web pages and, by extension, web applications, to change content dynamically without the need to reload the entire page. In practice, modern implementations commonly utilize JSON instead of XML.
7. **JAVA:** Java is a programming language and a platform. Java is a high level, robust, object-oriented and secure programming language.
8. **Technologies used for frontend development:** CSS, HTML, Bootstrap, jQuery, Ajax, JavaScript.
9. **Technologies used for Backend development:** Java, Struts2
10. **Database used:** MySQL

# **DATABASE STRUCTURE**

# 

**DFD**

**ADMIN Portal:**

DFD 0Level

Adding

**ADMIN**

Adding

DFD 1 Level

2

Filling details

3

Get access

4

dashboard

1

Get access

Enter Login Credentials

**ADMIN**

3

2

Filling details

**CSR Portal:**

DFD 0 Level

New Ticket

**CSR**

Assigning

DFD 1 Level

Assign Tech

Generate and

1

Get Access

Enter Login

dashboard

**CSR**

Credentials

Fill-up Details

**Tech Portal:**

DFD 0 Level

Update

View

Add

**Tech**

DFD 1 Level

Update

Enter Login

Get Access

Get Access

Dashboard

Credentials

**Tech**

Write

Get Access

**User Portal:**

DFD 0 Level

**User**

DFD 1 Level

Show

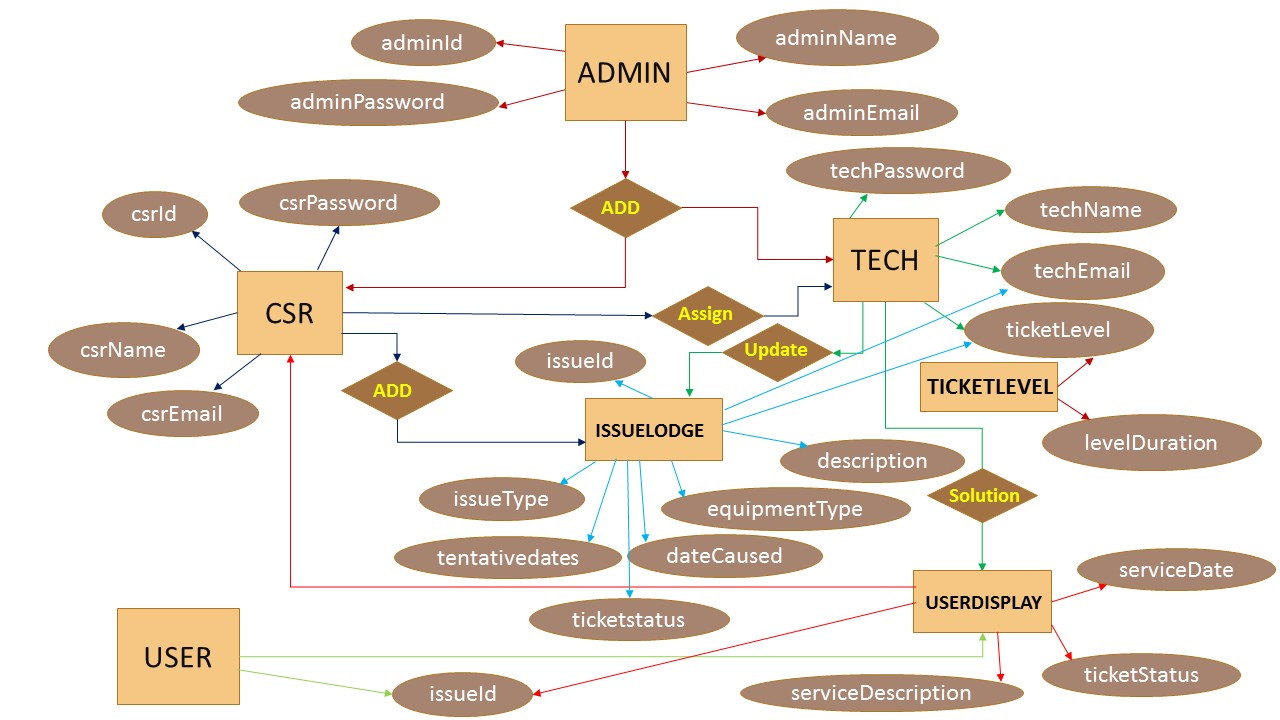
Enter ID

**Tracking ID**

**User**

Write

**ERD**

******

**CONCLUSION**

Online ticketing system computerizes nearly all of the process that goes behind resolving an issue. The main objective of this project is to manages the issues raised by the clients and save their time.

Without a ticketing system, keeping customer issues organized is difficult, which can create roadblocks to timely solutions. The ticket system for issue resolving tracks time spent on each step of the solution process. This keeps everything at a steady pace to ensure that our customers are getting taken care of quickly.

This project mainly includes flexibility, reducing manual work in an efficient manner, convenient, reliable, easily understandable and effective way to apply for their online ticket issue resolving.