**PATIENT TRACKING SYSTEM**

**ABSTRACT**

Main idea behind developing this application is to provide a solution for hospitals to manage patient’s records and details in single software. Patient tracking system application can manage information of patients health details, appointment details, payment details, past visiting details..etc. This application will be very useful to handling patient details. We can easily identified that one particular patient details.

Ward wise view is very helpful for We can easily identified where patient to be admitted. In this project have one remembered screen which is very helpful to give a medicine to patient on that time. This system focused on the patient record management for surgical department which include the patient record and patient admission record. The development of this system is to increase the function of patient record management. Besides that, it also to increase the security of patient record, minimize time in record calculation, and availability of record. The module that included in this system is patient record management, borrowing record, and record calculation. So, this computerized system will solve the problem that faced in the current manual system.

**1. INTRODUCTION**

This project objective is that who are need to see their patient they can find that patient details itself via this application. The visitors no need to wait in the long queue, and ask the details about the patient. This will be reduce the long time queue and save more time. The project is aimed to develop by **JAVA** as Front end and **MS SQL SERVER** as Back end. The back end is used to store the information in this system.

**1.1 SYSTEM SPECIFICATION**

**1.1.1 HARDWARE SPECFICATION:**

* Processor : P 4 700 GHz.
* RAM : 4GB RAM
* Hard Disk Drive : 40 GB HDD

**1.1.2 SOFTWARE SPECIFICATION:**

* + Operating System : Windows XP/7/8/10
  + Front End : JAVA
  + Back End : MY SQL

1. **SYSTEM STUDY**

**2.1 EXISTING SYSTEM:**

The existing system is bus pass management is very hard to manage manually. In this system passenger should take the pass anywhere we go. It may sometime chance to miss the pass. Passenger only responsible for the bus pass. This may hard to use this system which is have some drawbacks

**2.1.1 DRAWBACKS:**

The existing system has the following drawbacks.

* Waiting a long time in queue.
* Getting some irritation from the nurse response.
* We can’t track the patient manually.

**2.2 PROPOSED SYSTEM:**

This tracking system will helps to tracking the patient as room wise. Attender directly find the patient room and patient details. Doctors also use this application which will be very useful in the hospital. Every one can be know about the patient details.

**2.2.1 FEATURES:**

* No need to waiting in queue
* Manpower reduce
* Time saving

1. **SYSTEM DESIGN AND DEVELOPMENT**

**3.1 FILE DESIGN**

The selection of the file system design approach is done according to the needs of the developers what are the needed requirements and specifications for the new design. It allowed us to identify where our proposal fitted in with relation to current and past file system development. Our experience with file system development is limited so the research served to identify the different techniques that can be used. The variety of file systems encountered show what an active area of research file system development is. The file systems may be from one of the two fundamental categories. In one category, the file system is developed in user space and runs as a user process. Another file system may be developed in the kernel space and runs as a privileged process. Another one is the mixed approach in which we can take the advantages of both aforesaid approaches. Each development option has its own pros and cons. In this article, these design approaches are discussed.

**3.2 INPUT DESIGN**

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:’

* What data should be given as input?
* How the data should be arranged or coded?
* The dialog to guide the operating personnel in providing input.
* Methods for preparing input validations and steps to follow when error occur.

**OBJECTIVES**

* Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.
* It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.
* When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user
* will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow

**3.3 OUTPUT DESIGN**

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system’s relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.

2. Select methods for presenting information.

3. Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

* Convey information about past activities, current status or projections of the
* Future.
* Signal important events, opportunities, problems, or warnings.
* Trigger an action.
* Confirm an action.

**3.4 DATABASE DESIGN**

Today's businesses depend on their databases to provide information essential for day-to-day operations, especially in case of electronic commerce businesses who has a definite advantage with up-to-date database access. Good design forms the foundation of any database, and experienced hands are required in the automation process to design for optimum and stable performance.

Software Solutions have been constantly working on these platforms and have attained a level of expertise. We apply proven methodologies to design, develop, integrate and implement database systems to attain its optimum level of performance and maximize security to meet the client's business model.

### Business needs addressed:

* Determine the basic objects about which the information is stored
* Determine the relationships between these groups of information and the objects
* Effectively manage data and create intelligent information
* Remote database administration or on site administrative support
* Database creation, management, and maintenance
* Information retrieval efficiency, remove data redundancy and ensure data security

**3.5 SYSTEM DEVELOPMENT**

**3.5.1 DESCRIPTION OF MODULES**

1. Patient registration
2. Admission System
3. Search patient
4. Allocate Room

1.Patient registration:

In this module nurse will register patient detail before admit the patient. Which is very helpful to track patient details.

2. Admission system

An admission system module is used to admit the patients in room wise. So we can identify the patient as room wise and also we can manage what the rooms are available.

3. Search patient

We can search the patient as room wise and patient wise. In this patient should be register in patient registration module.

1. Allocate room

It will be automatically allocate the patient room when the patient are admitting. Which is very useful to allocate the room

1. **SYSTEM TESTING AND IMPLEMENTATION**

**SYSTEM TESTING**

System testing is the process of exercising software with the intent of finding and ultimately correcting errors. This fundamental philosophy does not change for web applications, because Web-based systems and application reside on a network and interoperate with many different operating system, browsers, hardware platforms, and communication protocols; the search for errors represents a significant challenge for web application.

The distributed nature of client\server environments, the performance issues associated with transaction processing, the potential presence of a number of different hardware platforms, the complexities of network communication, the need to serve multiple clients from a centralized database and the requirements imposed on the server all combine to make testing of client\server architectures.

System testing is actually a series of different tests whose primary purpose is to fully exercise the computer based system. System testing is the state of implementation that is aimed at assuring that the system works accurately and efficiently. Testing is the vital to the success of the system. System testing makes the logical assumption that if all the parts of the system are correct, the goal will be successfully achieved.

**The objective of testing is as follows:**

* + Testing is the process of executing a program with the intent of finding an error.
  + A successful test is that one of the cover of undiscovered error.

### TESTING ISSUES

* Client GUI considerations
* Target environment and platform diversity considerations
* Distributed database considerations
* Distributed processing considerations

**TESTING METHODOLOGIES**

System testing is state of implementation, which is aimed at ensuring that the system works accurately and efficiently as expect before live operation commences. It certifies that the whole set of programs hang together.

System testing requires a test plan that consists of several key activities and step for run program, string, system and user acceptance testing. The implementation of newly designed package is important in adopting a successful new system

Testing is the important stage in software development. the system test in implementation stage in software development process. The system testing implementation should be confirmation that all is correct and an opportunity to show the users that the system works as expected. It accounts the largest percentage of technical effort in the software development process.

Testing phase in the development cycle validates the code against the functional specification testing is vital to achievement of the system goals. The objective of the testing is to discover errors to fulfills this objective a series of test step unit, integration. validation and system tests were planned and executed the test steps are:

**System Testing**

Testing is an important phase in project development. System testing makes a logical assumption that if all parts of the system are correct, and the goal will be achieved successfully. The software must meet the user specification and it must satisfy according to the needs of the users.

Testing is the process of executing a project within the intend of finding errors. A good test case is one that has a high probability of finding an undiscovered error.

**Unit Testing**

Unit testing focuses verification efforts on the smallest unit of software design of the module. This is also known as “module testing”. This testing is carried out during programming stage itself. In this testing step, each module is found to be working satisfactorily as regards to the expected output of the modules.

**In Project**, Each module such customer registration module, request module, employee details module, stock module, vehicle module and area detail modules are tested individually for example, Customer details module can contain the more forms to maintain the information so all forms could be tested like entered information store appropriately in database access page or not. If correctly accessed means the testing of registration module successfully completed. Likewise all modules are tested successfully.

**Integration Testing**

Data can be lost across an interface, one module can have adverse effect on another sub function when combined it may not produce the desired major functions. Integration testing is a systematic testing for constructing test to uncover errors associated within an interface.

The objectives taken from unit tested modules and a program structure is built for integrated testing. All the modules are combined and the test is made.

A correction made in this testing is difficult because the vast expenses of the entire program complicated the isolation of causes. In this integration testing step, all the errors are corrected for next testing process.

**In Project,** Integration of two modules can be tested together such as customer registration details and customer login module for verification purposes providing proper accessibility to users. The communication of Registration and Login module can test and executed successfully.

**Validation Testing**

After the completion of the integrated testing, software is completely assembled as a package; interfacing error has been uncovered and corrected and a final series of software test validation begins.

Validation testing can be defined in many ways but a simple definition is that validation succeeds when the software function in a manner that can be reasonably expected by the customer. After validation test has been conducted, one of two possible conditions exists:

**In this project,** Admin login details form Enter without username and password in textbox enter the submit button then Login failed message otherwise checks the both textbox value that is true means valid page displayed. Enter Password Displaying password character \*.if it displays the characters security is not availed so testing of software is failed.

**Output Testing**

The next process of validation testing, is output testing of the proposed system, since no system could be successful if it does not produce the required output in the specified format. Asking the user about the format required, list the output to be generated or displayed by the system under considerations.

Output testing is a different test whose primary purpose is to fully exercise the computer based system although each test has a different purpose all the work should verify that all system elements have been properly integrated and perform allocated functions.

The output format on the screen is found to be corrected as the format was designed in the system design phase according to the user needs for the hard copy also; the output testing has not resulted in any correction in the system.

**In project** All the forms are tested as it gives the necessary output to the user’s search such as view response details.

1. **CONCLUSION**

The aim of this project is that in the hospital we no need to wait for knowing patient details. We can track the patient itself. Just give patient mobile number and we can track it.

**FUTURE ENHANCEMENT**

This application is developed by using JAVA and MS SQL SERVER as back end.In this future system will be add more features like discharge, lab report etc it’s more help to see all the details via this application, we can took all the details about the patient and tracking, its may effect more cost effective time estimation. We can try to complete the all these module by this application.

**BIBLIOGRAPHY**

* **Flanagan96**  
  David Flanagan, Java in a Nutshell, second ed., O'Reilly, 1996.
* **Flanagan99**  
  David Flanagan, Java Foundation Classes in a Nutshell, O'Reilly, 1999.
* **Gea99a**  
  David M. Geary, Graphic Java 2: Mastering the JFC, vol. I, AWT, third ed., Sun Microsystems Press, 1999.

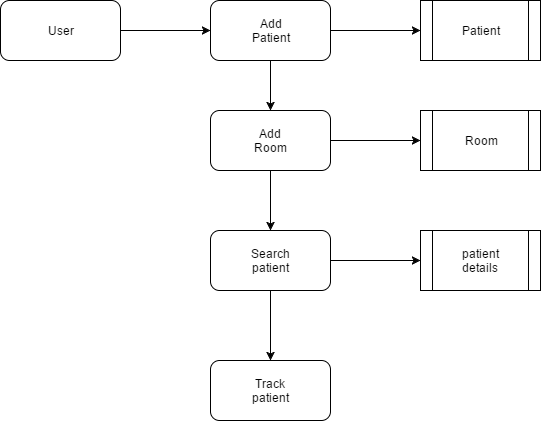
**APPENDICES**

1. **DATA FLOW DIAGRAM**

LEVEL 0:

****

LEVEL 1:

****

1. **TABLE STRUCTURE**

**TABLE NAME : ADMIN**

**PRIMARY\_KEY : ID**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **DESCRIPTION** |
| Id | Int |  | Admin id |
| Username | Varchar | 30 | Admin username |
| password | Varchar | 30 | Admin password |

**TABLE NAME : PATIENT**

**PRIMARY KEY : ID**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **DESCRIPTION** |
| Id | Int | 10 | Patient id |
| Name | Varchar | 30 | Application name |
| Address | Varchar | 30 | Address |
| Contactno | Varchar | 10 | Contact number |
| Emergency | Varchar | 10 | Emergency num |
| Age | Int | 10 | age |

**TABLE NAME : ADMISSION**

**PRIMARY KEY : ADMISSIONID**

**FOREIGN KEY : UID**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **DESCRIPTION** |
| Admissionid | Int | 10 | Admission id |
| Uid | Int | 10 | User id |
| Floor | Varchar | 30 | Floor name |
| Room no | Varchar | 30 | Room number |
| Doctor | Varchar | 10 | Doctor |
| Disease | Varchar | 10 | Disease |

1. **SAMPLE CODING**

<html>

<head>

<link href="/bootstrap/bootstrap.css" rel="stylesheet"/>

<link href="/bootstrap/datatables.css" rel="stylesheet"/>

<link rel="stylesheet" type="text/css" href="/home/login.css">

</head>

<body>

<div class="login-reg-panel">

<div class="white-panel">

<div class="login-show">

<h2>Patient Tracking System</h2>

<input id="username" type="text" placeholder="Username">

<input id="password" type="password" placeholder="Password">

<input onclick="login()" type="button" value="Login">

</div>

</div>

</div>

</body>

<script src="/bootstrap/jquery.min.js"></script>

<script src="/home/login.js"></script>

<script src="/bootstrap/datatable.min.js"></script>

<script src="/bootstrap/datatable.bootstrap.min.js"></script>

<script src="/bootstrap/bootstrap.min.js"></script>

<script>

function login() {

var username = $("#username").val();

var password = $("#password").val();

$.ajax({

type: "GET",

url: "/api/admin\_login/" + username + "/" + password

}).done(function (data) {

console.log(data);

if (data.length == 0) {

alert('Invalid username or password please enter valid username and password')

} else {

window.location = "/home";

}

});

}

</script>

</html>

<html>

<head>

<link href="/bootstrap/bootstrap.css" rel="stylesheet"/>

<link href="/bootstrap/datatables.css" rel="stylesheet"/>

<link rel="stylesheet" type="text/css" href="/home/home.css">

</head>

<body>

<div class="container">

<nav class="main-navigation">

<div class="navbar-header animated fadeInUp">

<a class="navbar-brand" href="#">Patient Tracking System</a>

</div>

<ul class="nav-list">

<li class="nav-list-item">

<a href="#htab1" data-toggle="tab" class="nav-link">Add Rooms</a>

</li>

<li class="nav-list-item">

<a href="#htab2" data-toggle="tab" class="nav-link">Add Patients</a>

</li>

<li class="nav-list-item">

<a href="#htab3" data-toggle="tab" class="nav-link">Patient Details</a>

</li>

<li class="nav-list-item">

<a href="#htab4" data-toggle="tab" class="nav-link">Admit Room</a>

</li>

<li class="nav-list-item">

<a href="#htab5" data-toggle="tab" class="nav-link">Track Patients</a>

</li>

<li class="nav-list-item">

<a href="/" class="nav-link">Logout</a>

</li>

</ul>

</nav>

<div class="row">

<div class="col-sm-12">

<div class="tab-content" style="margin: 30px">

<div role="tabpanel" class="tab-pane fade in active" id="htab1">

<h1>Add Rooms</h1>

<div class="col-sm-4">

<div class="form-group">

<label>Floor No:</label>

<input id="floor1" type="text" class="form-control">

</div>

<div class="form-group">

<label>Room No:</label>

<input id="room1" type="text" class="form-control">

</div>

<button onclick="addRoom()" type="button" class="btn btn-primary btn-md">Add Room</button>

</div>

</div>

<div role="tabpanel" class="tab-pane fade" id="htab2">

<h1>Add Patient</h1>

<div class="col-sm-4">

<div class="form-group">

<label>Patient Name:</label>

<input id="name2" type="text" class="form-control">

</div>

<div class="form-group">

<label>Mobile Number:</label>

<input id="mobile2" type="text" class="form-control">

</div>

<div class="form-group">

<label>Emergency Number:</label>

<input id="emergency2" type="text" class="form-control">

</div>

<div class="form-group">

<label>Age:</label>

<input id="age2" type="text" class="form-control">

</div>

<div class="form-group">

<label>Gender:</label>

<select class="form-control" id="gender2">

<option>Male</option>

<option>Female</option>

<option>Transgender</option>

</select>

</div>

<div class="form-group">

<label>Address:</label>

<input id="address2" type="text" class="form-control">

</div>

<button onclick="addPatient()" type="button" class="btn btn-primary btn-md">Add Patient</button>

</div>

</div>

<div role="tabpanel" class="tab-pane fade in" id="htab3">

<h1>Patient Details</h1>

<table class="table table-dark">

<thead>

<tr>

<th scope="col">#</th>

<th scope="col">Patient Id</th>

<th scope="col">Mobile Number</th>

<th scope="col">Patient Name</th>

<th scope="col">Emergency Number</th>

<th scope="col">Age</th>

<th scope="col">Gender</th>

</tr>

</thead>

<tbody id="patient\_body">

</tbody>

</table>

</div>

<div role="tabpanel" class="tab-pane fade in" id="htab4">

<h1>Admit Room</h1>

<div class="col-sm-4">

<div class="form-group">

<label>Mobile Number:</label>

<input id="mobile4" onchange="getPatientDetailsByNumber()" type="text" class="form-control">

</div>

<div class="form-group">

<label>Floor:</label>

<select class="form-control" id="floor4">

</select>

</div>

<div class="form-group">

<label>Room:</label>

<select class="form-control" id="room4">

</select>

</div>

<div class="form-group">

<label>Doctor Name:</label>

<select class="form-control" id="doctor4">

</select>

</div> <div class="form-group">

<label>Disease:</label>

<select class="form-control" id="disease4">

</select>

</div>

<div class="form-group">

<label>Patient Name:</label>

<input id="name4" type="text" class="form-control" disabled>

</div>

<div class="form-group">

<label>Emergency Number:</label>

<input id="emergency4" type="text" class="form-control" disabled>

</div>

<div class="form-group">

<label>Age:</label>

<input id="age4" type="text" class="form-control" disabled>

</div>

<div class="form-group">

<label>Gender:</label>

<input id="gender4" type="text" class="form-control" disabled>

</div>

<div class="form-group">

<label>Address:</label>

<input id="address4" type="text" class="form-control" disabled>

</div>

<button onclick="addAdmission()" type="button" class="btn btn-primary btn-md">Admit Room

</button>

</div>

</div>

<div role="tabpanel" class="tab-pane fade in" id="htab5">

<h1>Track Patient</h1>

<div class="col-sm-4">

<div class="form-group">

<label>Type Mobile Number:</label>

<input id="mobile5" type="text" class="form-control">

</div>

<div class="form-group">

<button onclick="searchPatient()" type="button" class="btn btn-primary btn-md">Search</button>

</div>

<table class="table table-bordered">

<tbody id="track\_body">

</tbody>

</table>

</div>

</div>

</div>

</div>

</div>

</div>

</body>

<script src="/bootstrap/jquery.min.js"></script>

<script src="/home/home.js"></script>

<script src="/bootstrap/datatable.min.js"></script>

<script src="/bootstrap/datatable.bootstrap.min.js"></script>

<script src="/bootstrap/bootstrap.min.js"></script>

</html>

1. **SAMPLE INPUT**

1. **SAMPLE OUTPUT**