

篇培交走侑

# Building a Robust Hospital Management System

Presented by: Ravikant Turi, Infinite Computer Solutions

# Project Overview: Hospital Management System

Our project focuses on developing a comprehensive Hospital Management System designed to streamline administrative tasks, enhance patient care, and manage critical hospital operations efficiently. This presentation will detail the insurance administration module, a key component enabling flexible policy management for individuals and families.

# Insurance Module: Core Functionality

## Individual & Family Plans

Designed the capability to create and manage insurance policies for both individual patients and entire families, ensuring comprehensive coverage options.

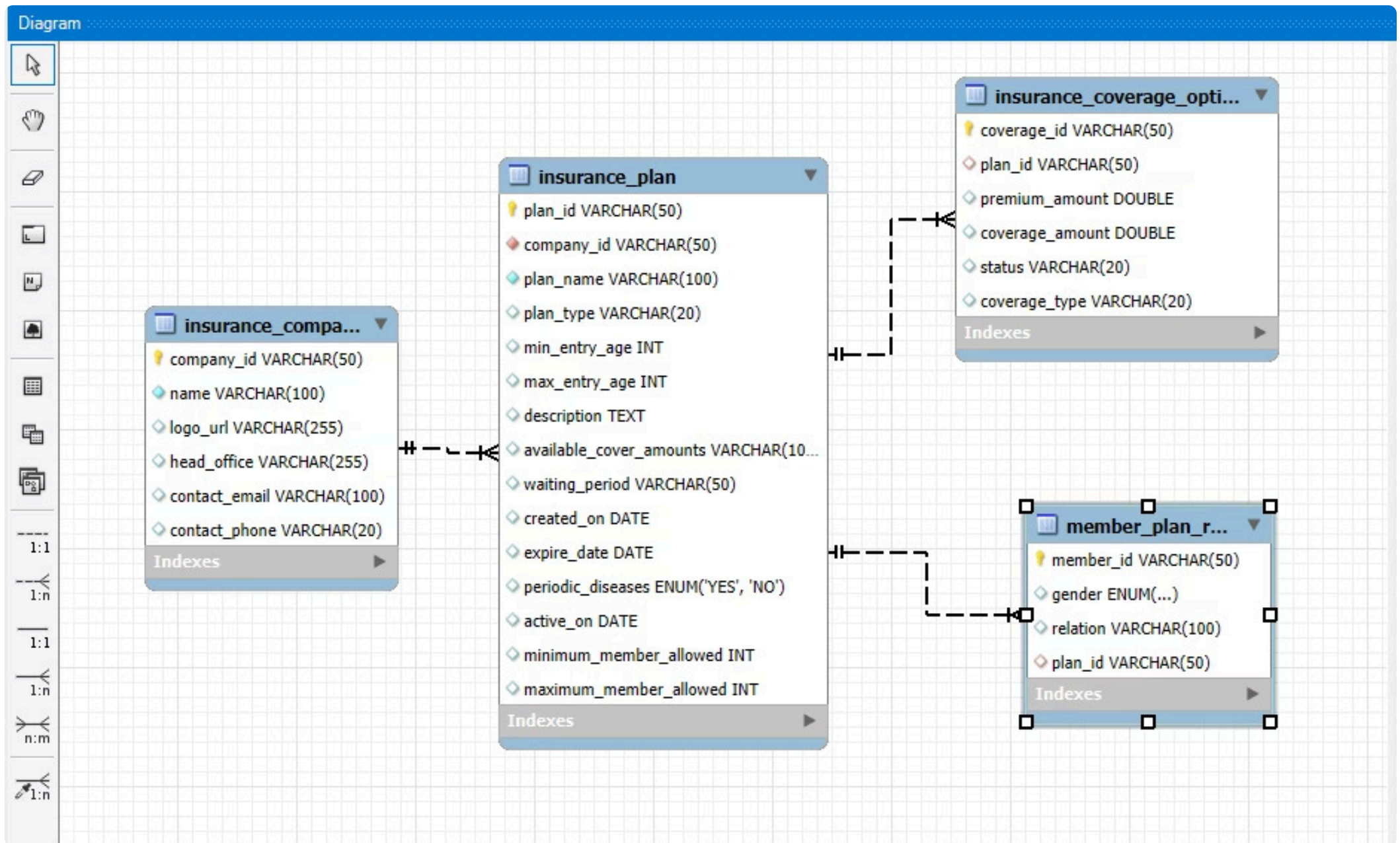
## Flexible Plan Configuration

Implemented a robust system allowing administrators to define various insurance plans with specific coverage tiers (Silver, Gold, Platinum).

## Membership Rules

Integrated features to set and enforce member rules for each insurance plan, dictating the number of individuals permitted under a family policy.

# Database Schema: Insurance Module



The insurance module's data structure is built upon four interconnected tables, ensuring data integrity and efficient retrieval of policy information. This normalized schema supports granular control over insurance companies, plans, coverage, and member rules.

# Architecture: MVC & Layered Design

The system adheres strictly to the Model-View-Controller (MVC) architectural pattern, promoting separation of concerns and maintainability. This layered approach enhances modularity, scalability, and ease of development.

## Model Layer

- Data logic and persistence
- Business rules implementation
- Hibernate 3.6 for ORM

## View Layer

- User interface representation
- JSF (JavaServer Faces) for frontend rendering

## Controller Layer

- Handles user input
- Interacts between Model and View
- Manages application flow

## Constants & Utilities

- Centralized configuration files
- Reusable utility functions
- Enhances code consistency

This structured approach significantly improves code organization and simplifies debugging processes.

# Technology Stack



## Java

Primary programming language for backend logic and application development, leveraging its robust ecosystem.



## MySQL

Relational database management system for storing and managing all system data, including patient records and insurance details.



## Hibernate 3.6

Object-Relational Mapping (ORM) framework for efficient database interaction and data persistence.



## Log4j

Logging framework used for debugging, error tracking, and monitoring application behavior in real-time.

# Admin Dashboard: Insurance Management

The administrator dashboard provides a centralized interface for managing all insurance-related data. This includes creating new insurance plans, modifying existing ones, and viewing all active policies.

<



# Validation

Add Insurance Plan

InsurancePlan Details

Company ID:

Insurance company is required.

\*Max Age:

0

Maximum age must be greater than 0.

\*Available Amount:

Cover amount is required.

\*MaximumMemberAllowed:

68,787,887

maximum 8 member is allowed

Periodic Diseases:

--Select--

Periodic diseases field is required.

Plan Name:

Plan name is required.

\*Min Age:

0

Minimum age must be greater than 0.

\*Waiting Period (Month):

Invalid waiting period.

\*minimumMeberAllowed:

9

minimum cant be greater than maximum

\*DURATION:

--Select--

You must choose a duration.

\*PlanType:

FAMILY

\*Description:

Description is required.

\*Active On:YYYY-MM-DD

Active On date is required.

chose Members

# Key Features: Dashboard Usability

## 1 Efficient Data Display

All insurance plans are displayed in a clear, tabular format, providing a quick overview of available policies and their attributes.

## 2 Dynamic Sorting

Implemented sorting capabilities for various columns (e.g., plan name, company, coverage level) to enable quick data retrieval and analysis.

## 3 Seamless Pagination

Designed pagination controls to manage large datasets, ensuring optimal performance and user experience when browsing numerous insurance plans.

# Conclusion & Future Scope

- The insurance administration module successfully provides a robust and flexible solution for managing diverse insurance plans within the Hospital Management System.
- Future enhancements could include integration with external insurance providers' APIs and advanced reporting features for policy utilization.
- This project demonstrates a solid foundation for scalable and maintainable enterprise applications.