

Design: Database

Adam Fleener

Overview:

I intend to utilize MongoDB to hold the data for the application. This will be a helpful option moving in the development of the project tracking app. The free cloud option available for my application will reduce the cost to production and will minimize maintenance costs in the future.

The application will utilize REST API for the application to communicate and assist with the data transfer and interactions with the user.

Database Design:

Contacts and clients will be related to each other as the contact could potentially become a client.

Contact Table:

```
{ id: string,  
  Name: string  
    { FirstName: string  
      Last Name: string }  
}
```

Email: string,

Phone: string,

}

Clients:

```
{ id: string,  
  Name: string,  
  Address: string,  
  Description: string,  
}
```

Users:

The users are the approved users for the application and will have the information available to the update projects, tasks, and status of the project.

Document:

```
{ id: string,
```

FirstName: string,
LastName: string,
Role: string,
}

Tasks Document:

{ id: string,
TaskName: string,
TotalHours: string,
StatusID: string,
MilestoneID: string,
ProjectID: string,
EmployeeID: string,
}

Projects Document:

{ id: string,
ProjectName: string,
StartDate: string,
Rate: string,
Budget: string,
Active: string,
TotalHours: string,
LaborCosts: string,
MaterialCosts: string,
TotalCosts: string,
ProjectManagementID: string,
StatusID: string,
ClientID: string
}

Project Status Document:

```
{ id: string,  
  Status: string  
}
```

Task Status Document:

```
{ id: string,  
  Status: string }
```

Budget:

The final portion of the database design will involve the costs associated to the project. Below is the design for the Budgetary costs as it relates to the application design.

Milestone Status Document:

```
{ id: string,  
  Status: string }
```

Milestones Document:

```
{ id: string,  
  Name: string,  
  DueDate: string,  
  Deliverables: string,  
  StatusID: string,  
  ProjectID: string,  
  TotalHours: string  
}
```

Costs Document:

```
{ id: string,  
  Name: string,  
  Description: string,  
  PricePer: string,  
  Quantity: string,  
  TotalCost: string,  
  ProjectID: string,
```

MileStoneID: string

}

Hours Document:

{ id: string,

Date: string,

Time: string,

WorkCompleted: string,

TaskID: string,

ProjectID: string,

}

This design will show the relationship between the tables and provide the user an opportunity to update the information as needed.

Notes:

Nothing has changed.