Out-World Space Camp
Reservation System

Tyler Hansen | fsdp.tylerdhansen.com | hansen.tyler3@outlook.com | 08/06/2020

About Me

Growing up, I had always been the technologically adept person in my family. Any time my parents had an issue with a piece of technology they would call for me to fix it. Over time, my interest in technology grew and in my free time I had begun to dabble in video game design using Unity and realized that I really enjoyed coding. I always enjoyed stories when I was younger and to me, code tells the story of an application.

I ended up at Centriq because although I enjoyed teaching myself languages like C++ and Java, I realized that I wanted an even deeper dive into coding than my resources could provide. I have loved my time here at Centriq and have learned so much more than I ever thought was possible in such a short amount of time. I learned that I love seeing an application slowly start to come together from the smallest of details to the most complicated pieces of functionality.
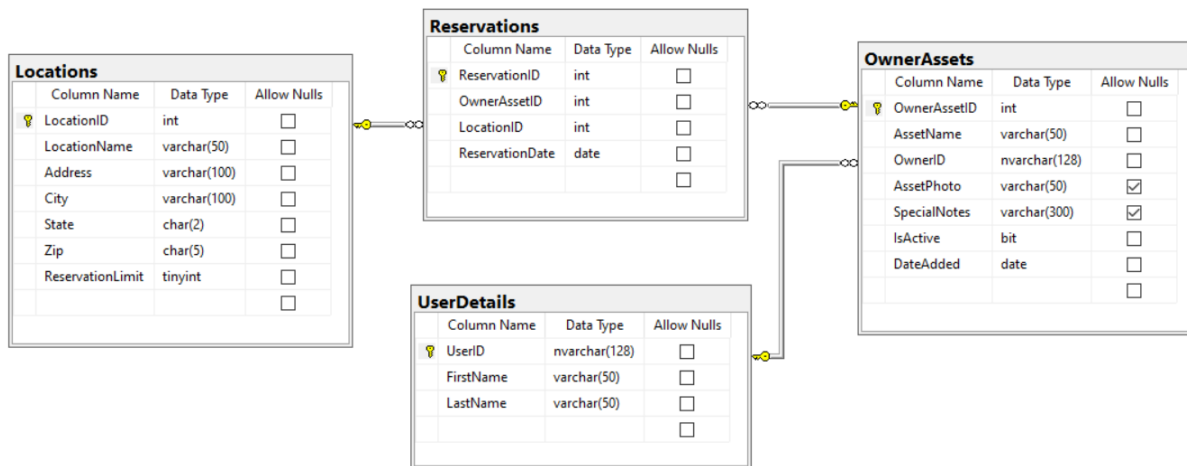
I chose to do the reservation system as my final project because having worked for multiple movie theaters when I was younger, and it reminded me of the point of sale systems we used. Working on this project, I learned how to pace myself when given a set of requirements for an application and I even learned a lot about where my strengths were in the process. I am thrilled to present my final project and show off its functionality.

Project Requirements

The requirements for this project were to build an application where a user could create an account, add an asset, and make a reservation for a service for their asset. Admins are the only ones with CRUD functionality for the locations of the company, they can view all reservations and assets, override reservations limits and place reservations manually, as well as update any notes. Employees have the capability of viewing locations, assets, and reservations, but can only edit the notes of assets and change reservation information. Users have the ability to create asset

profiles including the ability to upload a photo, place reservations for their assets and have full CRUD functionality for their assets and reservations.
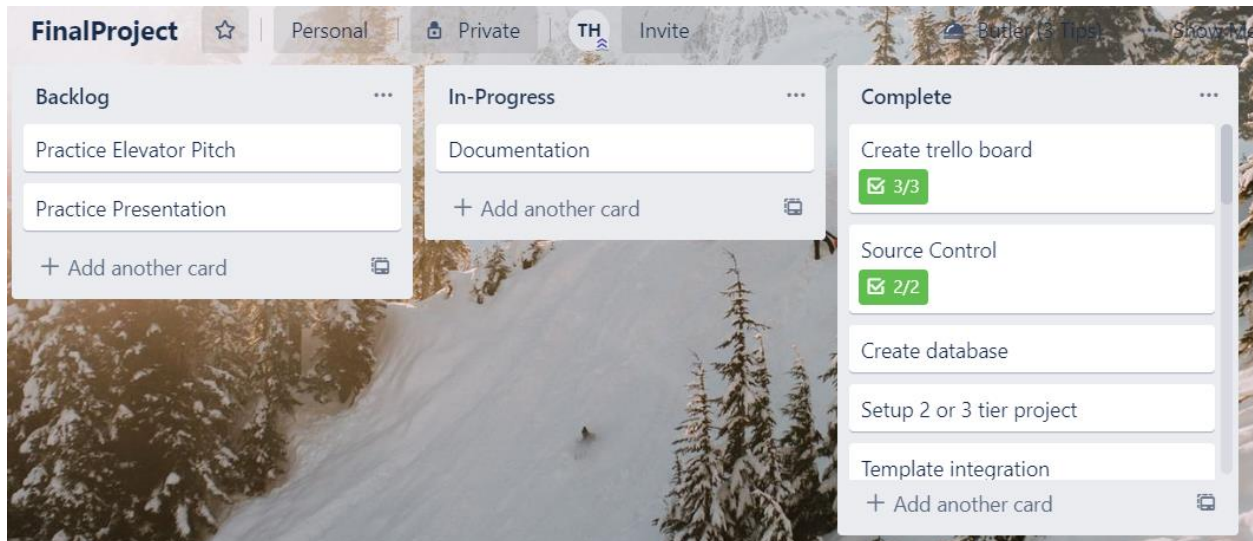
SQL Schema

Reservations

Column Name	Data Type	Allow Nulls
ReservationID	int	☐
OwnerAssetID	int	☐
LocationID	int	☐
ReservationDate	date	☐
		☐

Locations

Column Name	Data Type	Allow Nulls
LocationID	int	☐
LocationName	varchar(50)	☐
Address	varchar(100)	☐
City	varchar(100)	☐
State	char(2)	☐
Zip	char(5)	☐
ReservationLimit	tinyint	☐
		☐

OwnerAssets

Column Name	Data Type	Allow Nulls
OwnerAssetID	int	☐
AssetName	varchar(50)	☐
OwnerID	nvarchar(128)	☐
AssetPhoto	varchar(50)	☑
SpecialNotes	varchar(300)	☑
IsActive	bit	☐
DateAdded	date	☐
		☐

UserDetails

Column Name	Data Type	Allow Nulls
UserID	nvarchar(128)	☐
FirstName	varchar(50)	☐
LastName	varchar(50)	☐
		☐

Use/Case Diagram

Controller	Admin	Employee	Client
Home			
Index	x	x	x
Locations			
Index	x	x	x
Details	x		
Create	x		
Edit	x		
Delete	x		
Children			
Index	x	x	x
Details	x	x	x
Create	x		x
Edit	x	x (only notes)	x
Delete	x		x
Reservations			
Index	x	x	x
Details	x	x	x
Create	x		x
Edit	x	x	x
Delete	x		x

Workflow Process

While working on my project, I used a Trello board to help me manage the workload and ensure that I was keeping track of my progress. This was imperative as I was able to verify that I had all components of the project done.

Coding Snippets

This snippet of code was used to prevent clients from seeing reservations that they did not create themselves.

```csharp
// GET: Reservations
public ActionResult Index()
{
    if (User.IsInRole("Admin") || User.IsInRole("Employee"))
    {
        var reservations = db.Reservations.Include(r => r.Location).Include(r => r.OwnerAsset);
        return View(reservations.ToList());
    }
    else
    {
        string currentUser = User.Identity.GetUserId();

        var reservations = db.Reservations.Where(x => x.OwnerAsset.OwnerID == currentUser);
        return View(reservations.ToList());
    }
}
```

Coding Snippets Continued

This snippet of code is used to enforce the reservation limit for locations.

```csharp
// POST: Reservations/Create
// To protect from overposting attacks, please enable the specific properties you want to bind to, for
// more details see https://go.microsoft.com/fwlink/?LinkId=317598.
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult Create([Bind(Include = "ReservationID,OwnerAssetID,LocationID,ReservationDate")] Reservation reservation)
{
    if (ModelState.IsValid)
    {
        var place = db.Locations.Where(x => x.LocationID == reservation.LocationID).FirstOrDefault();
        var dateCount = db.Reservations.Where(x => x.ReservationDate == reservation.ReservationDate && x.LocationID == reservation.LocationID).Count();

        int reserveLimit = place.ReservationLimit;
        if (reserveLimit > dateCount)
        {
            db.Reservations.Add(reservation);
            db.SaveChanges();
            return RedirectToAction("Index");
        }
        else
        {
            return View("LimitExceeded");
        }
    }

    ViewBag.LocationID = new SelectList(db.Locations, "LocationID", "LocationName", reservation.LocationID);
    ViewBag.OwnerAssetID = new SelectList(db.OwnerAssets, "OwnerAssetID", "AssetName", reservation.OwnerAssetID);
    return View(reservation);
}
```

This snippet of code is used to dynamically grab the parent information for an asset whenever it is created by a client, automatically set their IsActive property to true, and grab the date in which the asset was created.

```csharp
//Dynamically grabs current logged in user and sets it as the OwnerID if the user is a client
if (User.IsInRole("Client"))
{
    string currentUser = User.Identity.GetUserId();

    ownerAsset.OwnerID = currentUser;
}

//Automatically sets the IsActive property to true
ownerAsset.IsActive = true;

//Dynamically adds the current date
ownerAsset.DateAdded = DateTime.Now;
```

Tyler Hansen

FULL-STACK DEVELOPER

816-785-1746
hansen.tyler3@outlook.com
www.tylerdhansen.com

Qualifications

- Solid foundational knowledge of designing and developing full-stack web applications using .NET framework.
- Excellent customer relations
- Solid troubleshooting and time management skills
- Quick at learning new skills

Technical Skills

Front End:	HTML5, JavaScript, jQuery, jQueryUI, CSS3, Responsive/Mobile Web Development, Bootstrap, ReactJS
Middle Tier:	Visual Studio, C#.NET, ASP.NET, LINQ, MVC, EF
Back End:	ADO.NET, SQL, SQL Server, SSMSE

Independent Development Projects

- **Personal Site**: www.tylerdhansen.com
- **StoreFront**: Created a secure application for managing product data. Application is built to simulate an online store front with a shopping cart. Administrators have the ability to manage product, category and vendor data.
- **S.A.T. Scheduling Administration Tool:** Created a secure application for managing product data. Application is built to simulate an online class scheduling system. Administrators will have the ability to manage students, courses, scheduled classes, and enrollments.
- **Final Project**: Created a secure data-driven ASP.NET MVC application from design through deployment for managing the tracking and organization of hardware and software within a company. Administrators have the ability to manage employee, department data and all details relating to assigned hardware and software.

Technical Training

CENTRIQ TRAINING | KANSAS CITY, MO   **April 27th – PRESENT**
FULL-STACK WEB DEVELOPER PROGRAM

Core Competencies:

- MVC Framework
- Trouble Shooting & Debugging
- Source Control
- Agile/Scrum (Created Team Project)
- Website Deployment
- Pair Programming
- Code Review
- Professionalism, Teamwork, Problem Solving & Effective Communication

Professional Experience

AMC Theatres | Kansas City, MO   **October 2019 – February 2020**
Crew Member

- Ensured guest satisfaction
- Worked in concessions which included the duties of upselling and maintaining a clean environment
- Teamwork was crucial for maintaining the eb and flow of daily operations

Lutheran Senior Services | Columbia, MO   **July 2018 – July 2019**
Cook

- Worked in several different styles of kitchens ranging from short order to fine dining
- Interacted with residents on a daily basis while serving food as well as helping with any requests they had
- Attention to detail was a large part of day to day tasks as well as the ability to multitask to ensure all duties were accomplished

Mediacom | Columbia, MO   **October 2017 – July 2018**
Broadband Specialist I

- Installed and repaired services at customer's homes
- Worked with people daily varying from trying to improve an irate customer's experience to building long lasting relationships with new customers
- Problem solving was key as majority of my time was spent correcting problems at a customer's house