Spair Bowling ALley REport

Josh Monaghan

Table of Contents

[Business Description 2](#_Toc184389667)

[Planning and Development 2](#_Toc184389668)

[Security Measures 3](#_Toc184389669)

[Technological Analysis 4](#_Toc184389670)

# Business Description

Spair Alley is a new bowling alley founded in the Newcastle area. The business offers arcade games, bowling, food and drinks, and is for all ages and families. After being open for a month, the business is struggling to keep track of all the customers and sessions. The aim of the database for Spair Alley is to manage a booking system that both staff and customers can use to book a bowling session. The database will need the following transactions:

* **Customers Session Creating:** Customers should have the ability to book sessions.
* **Creating a User Account:** Customers should have the ability to create an account.
* **Logging Into an Account:** Customers and staff should have the ability to log into their accounts to access features.
* **Update Booking:** Users should have the ability to update their bookings.
* **Cancel a Session:** Customers should have the ability to cancel a session.
* **Staff Account:** Staff should be able to create a separate account with a secret key to gain more access to the database.
* **Staff View:** Staff should be able to view/edit/cancel/add sessions and remove/view all the customers.

Bowling alley databases commonly hold all the transactions above. Depending on the need for the database, some bowling alleys will store scores and bowling groups. Under the circumstances of Spair Alley, the reason for the database is for booking which is why scores and groups aren’t included in the database. The groups that play at Spair Alley are not related to the bowling so including groups in the database is not needed. When deleting users, there are two options for what happens with their sessions. Option one is the session customer is set to null; this allows for it to still be in the database while the customer being removed. Option two is removing all the user’s sessions, this is the option Spair Bowling went with. It ensures that if a user is removed, there bookings are removed.

# Planning and Development

The structure of a database is an important part of designing and creating databases. The key factors are understanding what transactions are needed and what data needs to be stored. Spair Alley’s database is made up of two tables, Users and Sessions. Users stores all the customers information including: first name, last name, age, email, phone number, a hashed password, a secret code for the two factor and admin. Sessions is the bookings that are made, it includes: date and time, group amount, customer, and three nullable fields: first name, last name, phone number. The nullable fields are included for the reason of an admin entering a session for a user - it allows over the phone bookings to still be made without needing to create an account. For users logged in they don’t get the option to enter data into the nullable fields, allowing for those columns to be nullable. In sessions, the column customer is a foreign key that links to the users table, it ensures that the staff know what user booked the session and all the information they need to contact them. With this structure, the business requirements have been met effectively creating an efficient database for Spair Alley.

# Security Measures

Security is an important part of a database especially when handling Personal Identifiable Information. For this reason, security measures are outlined to ensure legal implications are followed and all data is secure from unauthorised individuals. Spair Alley has multiple security measures put in place to ensure no data is accessible to unauthorised individuals. Hashing is an important strategy to help with security. In Spair Alley, the user’s password is hashed and can never be unhashed. This allows for an extra layer of security ensuring that if the database was breached, the password would still be secure. N-Teir Layering is a technique very commonly used in industry, it separates the application into multiple layers. In Spair Alley, the database, logic and console application are all separate, this helps stop attackers from gaining access to everything if they attack a server. With N-Teir architecture included, if the console application was to be compromised by attackers, they still wouldn’t have access to the database server. Two Factor authentication is a commonly used technique when logging into or creating accounts. It adds an extra layer of security, ensuring that the person connecting is who they say they are. It works by providing the customer a QR Code to scan when creating an account and they use an authentication app like Google Authentication which will give them an OTP (One time passcode) which they enter into the console. This makes it more difficult for unauthorized individuals because they would need to compromise the customers phone to access their account.

To access a database, a connection string is needed. It includes the password, username, IP and database name to connect to it. Storing this into your source code is a huge vulnerability because if anyone gained access to that code, they could effortlessly access the database. Spair Alley uses aws secrets manager to overcome this issue. When the connection string is needed, it receives the aws credentials. These credentials are stored in the computers environment and requests the secrets which are then injected into the connection string. This way all credentials are removed from the source code. Using a secrets manager, instead of just environmental variables, allows the request to be monitored and analysed to ensure unauthorized individuals aren’t requesting the secrets. Input validation is incredibly important in industry, it checks the user’s inputs and if it’s a valid response. Parametrised queries is also an important security measure which ensures that queries can’t be SQL injected which could allow unauthorised individuals to access your database. Ensuring that all PII is securely stored and handled is important for both ethical and legal reasons. If PII wasn’t secured properly, it could lead to people not trusting the business or the business being fined or even shutdown.

# Technological Analysis

Security vulnerabilities are a huge issue in industry which is why it’s so heavily mandated. Vulnerabilities are faults in the source code that can be taken advantaged of to damage a business. Some common vulnerabilities are:

* **SQL Injection Attacks:** SQL injection is a strategy that takes advantage of non-parameterised queries by injecting malicious code into the user’s inputs. ([The 7 Biggest Database Security Risks and Threats, and How to Avoid Them | Buchanan Technologies](https://www.buchanan.com/database-security-risks-and-threats/?form=MG0AV3))
* **Ransomware Attacks:** Ransomware may encrypt your database making the data inaccessible. ( [The 7 Biggest Database Security Risks and Threats, and How to Avoid Them | Buchanan Technologies](https://www.buchanan.com/database-security-risks-and-threats/?form=MG0AV3))
* **Cross-Site Scripting:** Cross-site scripting is a common attack on the web that steals the session cookie to hijack the user’s session. ([Cross Site Scripting (XSS) | OWASP Foundation](https://owasp.org/www-community/attacks/xss/))
* **Encryption Flaws:** Encryption flaws is an exploit that uses bad encrypted data to gain access to sensitive data. ([A02 Cryptographic Failures - OWASP Top 10:2021](https://owasp.org/Top10/A02_2021-Cryptographic_Failures/))

These vulnerabilities are common attacks that still happen in the industry, understanding these attacks is the key to avoiding them. In Spair Alley, these vulnerabilities are resolved, for example:

* **SQL Injection Attacks:** These are avoided by using sanitised inputs and parametrised queries.
* **Ransomware Attacks:** These are avoided by backups being stored on site and off site weekly.
* **Cross-Site Scripting:** Cross-site scripting only effects web applications so this isn’t a vulnerability for console applications.
* **Encryption Flaws:** Encryption flaws are counteracted by using Argon2 which is a highly effective and secure hashing tool.

Using the security by design protocols allows for all these vulnerabilities to be avoided at the start of development. By using this protocol, it ensures that security is a focus and is constantly the biggest goal when it comes to securing a database. Ensuring that security measures are constantly updating reduces the risks of data breaches and vulnerabilities.