# CMPG223 Group Project Group 1

## Group Members:

* Alwyn Barnard 32033303
* Jadon Ockhuys 30415284
* Ryan Louw 30035023
* Henco Schutte 37823949
* Pirow Engelbrecht 37275119
* Smichael Coetzee 37246925

## Database Schema:

**Diagram

Description automatically generated**

## SQL statements used:

All create SQL statements:

CREATE TABLE [dbo].[Animal\_Type]( [Animal\_TypeID] [int] IDENTITY(1,1) NOT NULL, [AType\_Name] [nchar](50) NULL, [AType\_Endangered] [char](1) NULL, CONSTRAINT [PK\_Animal\_Type] PRIMARY KEY CLUSTERED ( [Animal\_TypeID] ASC )WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY] ) ON [PRIMARY] GO

REATE TABLE [dbo].[Animals]( [Animal\_ID] [int] IDENTITY(1,1) NOT NULL, [Animal\_Type] [int] NULL, CONSTRAINT [PK\_Animals] PRIMARY KEY CLUSTERED ( [Animal\_ID] ASC )WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY] ) ON [PRIMARY] GO ALTER TABLE [dbo].[Animals] WITH CHECK ADD CONSTRAINT [FK\_Animals\_Animal\_Type] FOREIGN KEY([Animal\_Type]) REFERENCES [dbo].[Animal\_Type] ([Animal\_TypeID]) ON UPDATE CASCADE ON DELETE CASCADE GO ALTER TABLE [dbo].[Animals] CHECK CONSTRAINT [FK\_Animals\_Animal\_Type] GO

CREATE TABLE [dbo].[Booking]( [Booking\_ID] [int] IDENTITY(1,1) NOT NULL, [Booking\_Date] [date] NULL, [Worker\_ID] [int] NOT NULL, [Booking\_Pay] [char](1) NOT NULL, [Guest\_Ari] [nchar](1) NOT NULL, [House\_ID] [int] NULL, [Guest\_ID] [int] NULL, [typeID] [int] NULL, CONSTRAINT [PK\_Booking] PRIMARY KEY CLUSTERED ( [Booking\_ID] ASC )WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY] ) ON [PRIMARY] GO ALTER TABLE [dbo].[Booking] WITH CHECK ADD CONSTRAINT [FK\_Booking\_Guests] FOREIGN KEY([Guest\_ID]) REFERENCES [dbo].[Guests] ([Guest\_ID]) ON UPDATE CASCADE ON DELETE CASCADE GO ALTER TABLE [dbo].[Booking] CHECK CONSTRAINT [FK\_Booking\_Guests] GO ALTER TABLE [dbo].[Booking] WITH CHECK ADD CONSTRAINT [FK\_Booking\_House\_Type] FOREIGN KEY([typeID]) REFERENCES [dbo].[House\_Type] ([TypeID]) GO ALTER TABLE [dbo].[Booking] CHECK CONSTRAINT [FK\_Booking\_House\_Type] GO ALTER TABLE [dbo].[Booking] WITH CHECK ADD CONSTRAINT [FK\_Booking\_Workers] FOREIGN KEY([Worker\_ID]) REFERENCES [dbo].[Workers] ([Worker\_ID]) ON UPDATE CASCADE ON DELETE CASCADE GO ALTER TABLE [dbo].[Booking] CHECK CONSTRAINT [FK\_Booking\_Workers] GO

CREATE TABLE [dbo].[Guests]( [Guest\_ID] [int] IDENTITY(1,1) NOT NULL, [Guest\_Name] [nchar](50) NULL, [Guest\_Surname] [nchar](50) NULL, [Guest\_Email] [nchar](100) NULL, CONSTRAINT [PK\_Guests] PRIMARY KEY CLUSTERED ( [Guest\_ID] ASC )WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY] ) ON [PRIMARY] GO

CREATE TABLE [dbo].[House]( [House\_num] [int] IDENTITY(1,1) NOT NULL, [House\_typeID] [int] NULL ) ON [PRIMARY] GO ALTER TABLE [dbo].[House] WITH CHECK ADD CONSTRAINT [FK\_House\_House\_Type] FOREIGN KEY([House\_typeID]) REFERENCES [dbo].[House\_Type] ([TypeID]) ON UPDATE CASCADE ON DELETE CASCADE GO ALTER TABLE [dbo].[House] CHECK CONSTRAINT [FK\_House\_House\_Type] GO

CREATE TABLE [dbo].[House\_Type]( [TypeID] [int] IDENTITY(1,1) NOT NULL, [Type\_Price] [float] NULL, [Type\_Name] [nchar](100) NULL, [Type\_Size] [int] NULL, CONSTRAINT [PK\_House\_Type] PRIMARY KEY CLUSTERED ( [TypeID] ASC )WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY] ) ON [PRIMARY] GO

CREATE TABLE [dbo].[Workers]( [Worker\_ID] [int] IDENTITY(1,1) NOT NULL, [Worker\_Name] [nchar](50) NULL, [Worker\_Surname] [nchar](50) NULL, [Worker\_StartDate] [date] NULL, [Worker\_Email] [nchar](100) NULL, CONSTRAINT [PK\_Workers] PRIMARY KEY CLUSTERED ( [Worker\_ID] ASC )WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY] ) ON [PRIMARY] GO

Delete SQL statements:

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

All update SQL statements:

Text

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Text

Description automatically generated

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with medium confidence

All insert SQL statements:

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

# Reports generated:

Graphical user interface, application

Description automatically generated

The report shown is generated by finding the guest’s email and sending the guest’s information back to the user.

Graphical user interface, application

Description automatically generated

The report shown here searches for when there is a booking open for a specific date and types of houses available on that date.

Graphical user interface, table

Description automatically generated

The report shown here shows a sum of available house types that are not booked for a specific date.

# User manual:

Getting started:

1. Go to “www.ParkieGrasLodge.com”.
2. Click on the download link.
3. Open setup file to start installing.
4. Choose Azure under database.
5. Install Azure if needed.
6. Finish installation.

Technical requirements:

* 16 GB RAM required.
* 5 GB disk space required.
* Intel core i7 or higher processer.
* 350-Watt power supply or higher.
* No GPU requirements.
* Windows 10 or newer version.

# Detail Diary:

Jadon Ockhuys:

* Improved diagrams from semester one – 1 day.
* GUI design – 1 day.
* Exception handling – 2 days.
* SQL Statements – 2 days.

Alwyn Barnard:

* Report – 3 hours.
* GUI design – 2 hours.
* Exception handling – 2 days.

Ryan Louw:

* Created databases – 1 day.
* ERD – 2 days.
* Programming – 1 week.

Pirow Engelbrecht:

* UML Diagrams – 1 day.
* SQL Statement for select – 1 day.

Henco Schutte:

* Filtering of databases – 4 days.
* GUI adjustments – 3 days.
* Programming – 1 week.

Smichael Coetzee:

* UML Diagrams – 1 day.