In [24]:
```python
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns # Visualization library
```

Data Pre-processing

A rigorous process of statistical analysis and data pre-processing is indispensable when choosing a dataset. This proposal outlines a systematic approach to scrutinize the dataset, handle anomalies, and transform features, ultimately laying a robust foundation for applying selected classifiers.

Before we proceed, we provide a description of what each of the features represent:

1. Age
2. Gender
3. BMI (Body Mass Index)
4. SBP (Systolic Blood Pressure)
5. DBP (Diastolic Blood Pressure)
6. FPG (Fasting Plasma Glucose)
7. FFPG (Final Fasting Plasma Glucose)
8. Cholestrol
9. Triglyceride
10. HDL (High-Density Lipoprotein)
11. LDL (Low-Density Lipoprotein)
12. ALT (Alanine Aminotransferase)
13. BUN (Blood urea nitrogen)
14. CCR (Creatinine Clearance)
15. Smoking Status
16. Drinking Status
17. Family History of Diabetes
18. Diabetes - **Class label**

Descriptive Statistics

We will begin by calculating descriptive statistics—namely, the mean, median, mode, variance, and standard deviation—to understand the central tendency, distribution spread, and shape of the features in our dataset

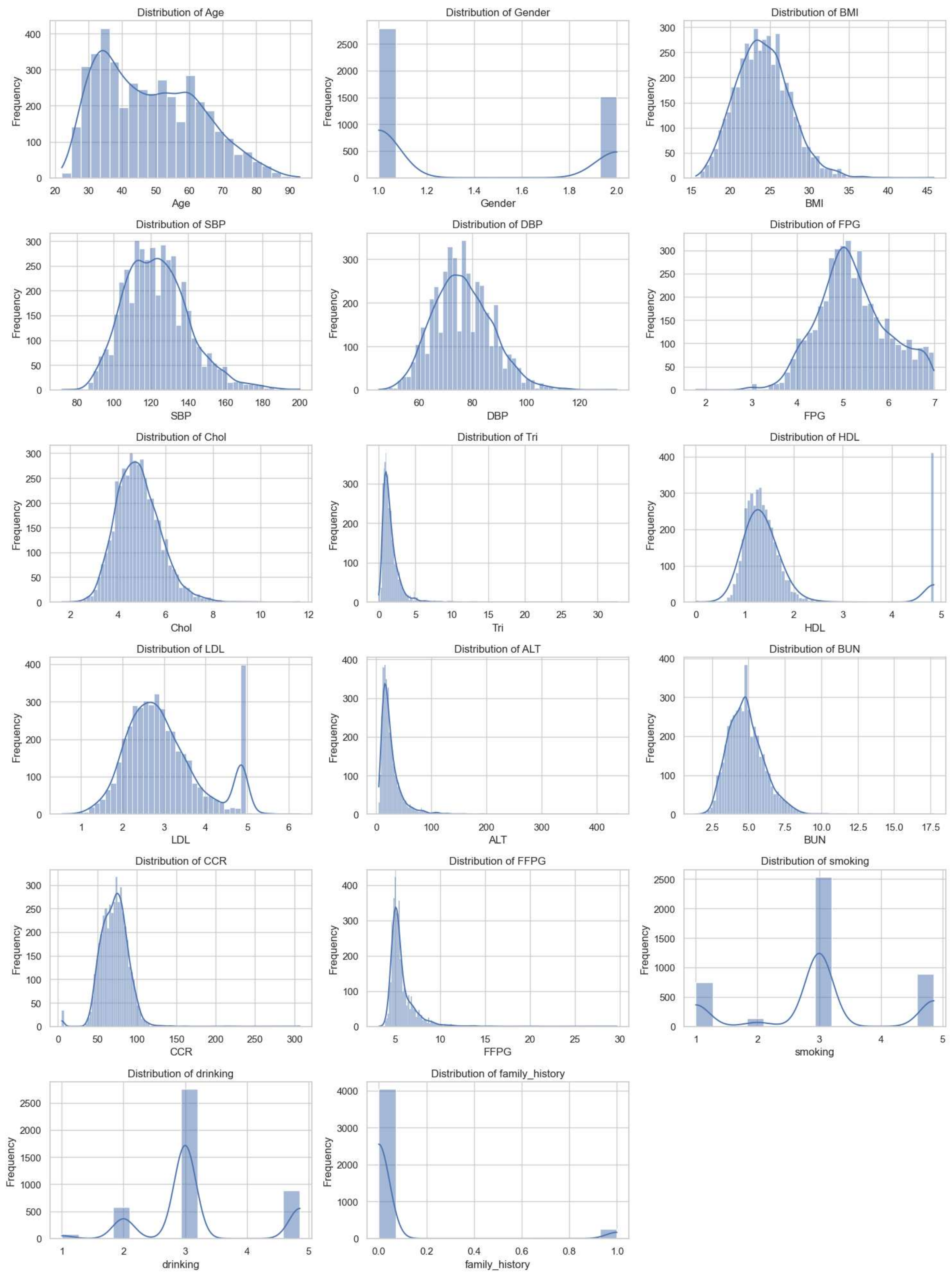
In [28]:
```
df = pd.read_csv(f"diabetes.csv")

# Compute the descriptive statistics for the dataset
summary = df.describe()

# Calculate Q1, Q3 and the IQR for each feature
Q1 = df.quantile(0.25)
Q3 = df.quantile(0.75)
IQR = Q3 - Q1

# Append Q1 and Q3 to the summary
summary.loc['Q1'] = Q1
summary.loc['Q3'] = Q3
summary.loc['IQR'] = IQR

summary
```

Out[28]:

Chol	Tri	HDL	LDL	ALT	BUN	CCR
4303.000000	4303.000000	4303.000000	4303.000000	4303.000000	4303.000000	4303.000000
4.860753	1.588476	1.666765	2.976220	26.754810	4.841088	71.557080
0.925377	1.260003	1.079377	0.892196	22.279922	1.251566	16.937394
1.650000	0.000000	0.000000	0.540000	4.500000	1.380000	4.860753
4.200000	0.860000	1.130000	2.350000	14.300000	3.960000	60.000000
4.790000	1.280000	1.340000	2.820000	20.500000	4.760000	72.000000
5.430000	1.940000	1.610000	3.420000	31.050000	5.570000	82.300000
11.650000	32.640000	4.860753	6.270000	436.200000	17.730000	307.000000
4.200000	0.860000	1.130000	2.350000	14.300000	3.960000	60.000000
5.430000	1.940000	1.610000	3.420000	31.050000	5.570000	82.300000
1.230000	1.080000	0.480000	1.070000	16.750000	1.610000	22.300000

Frequency Distribution of the features

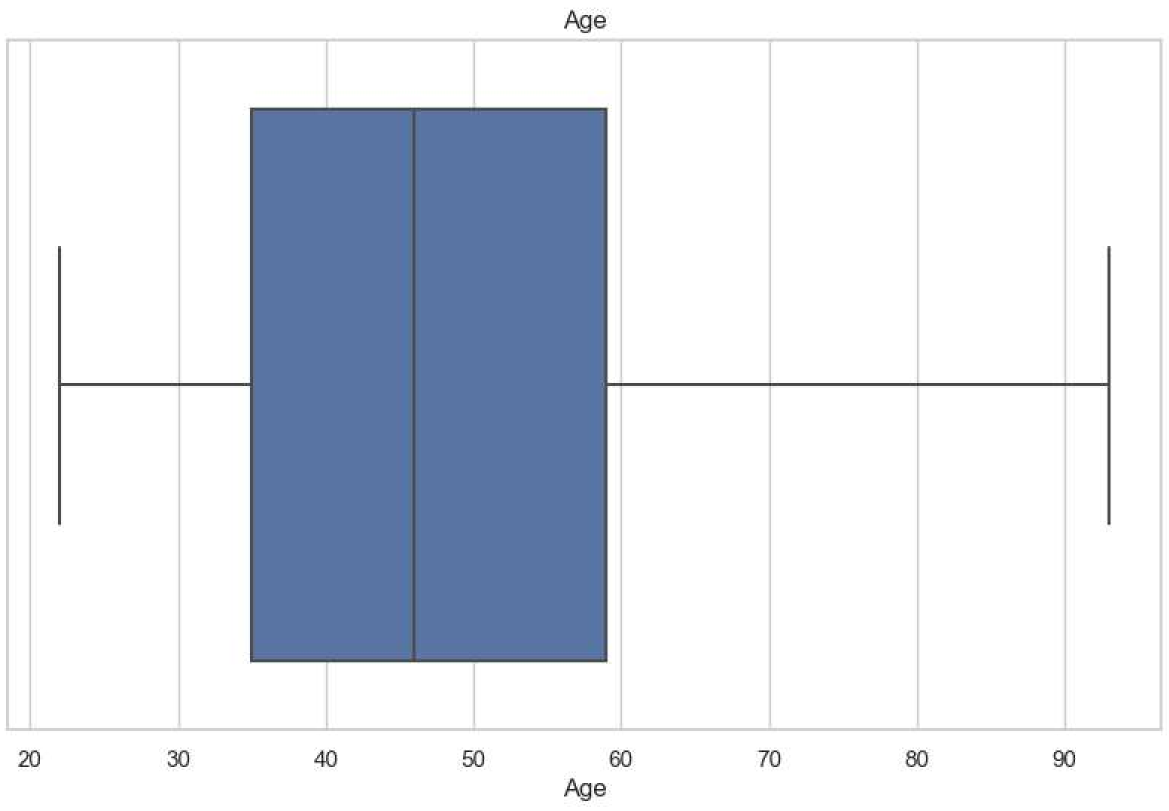
In [35]:
```
# Get the list of feature columns (excluding the class label)
feature_columns = df.columns[:-1]

# Create a 6x3 grid of distribution plots for features
plt.figure(figsize=(15, 20))
for i, feature in enumerate(feature_columns, 1):
    plt.subplot(6, 3, i)
    sns.histplot(df[feature], kde=True)
    plt.title(f'Distribution of {feature}')
    plt.xlabel(feature)
    plt.ylabel('Frequency')

plt.tight_layout()
plt.show()
```

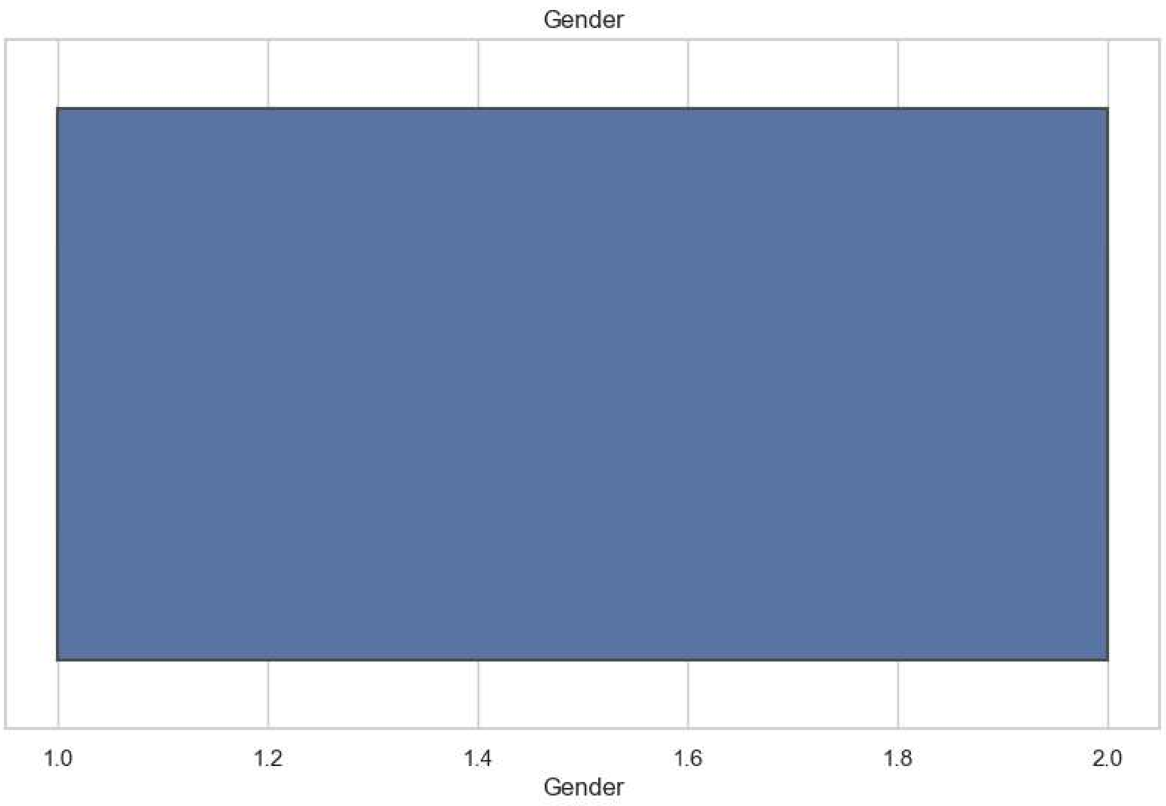


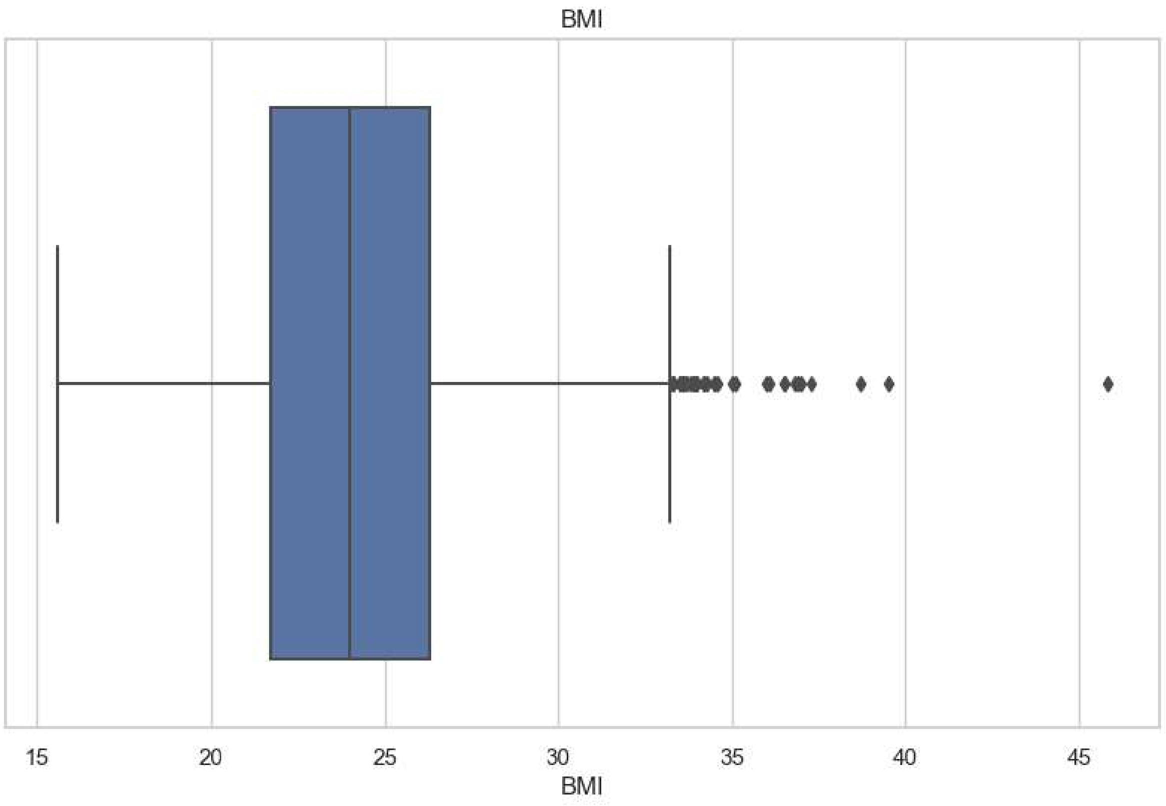
Outlier Analysis

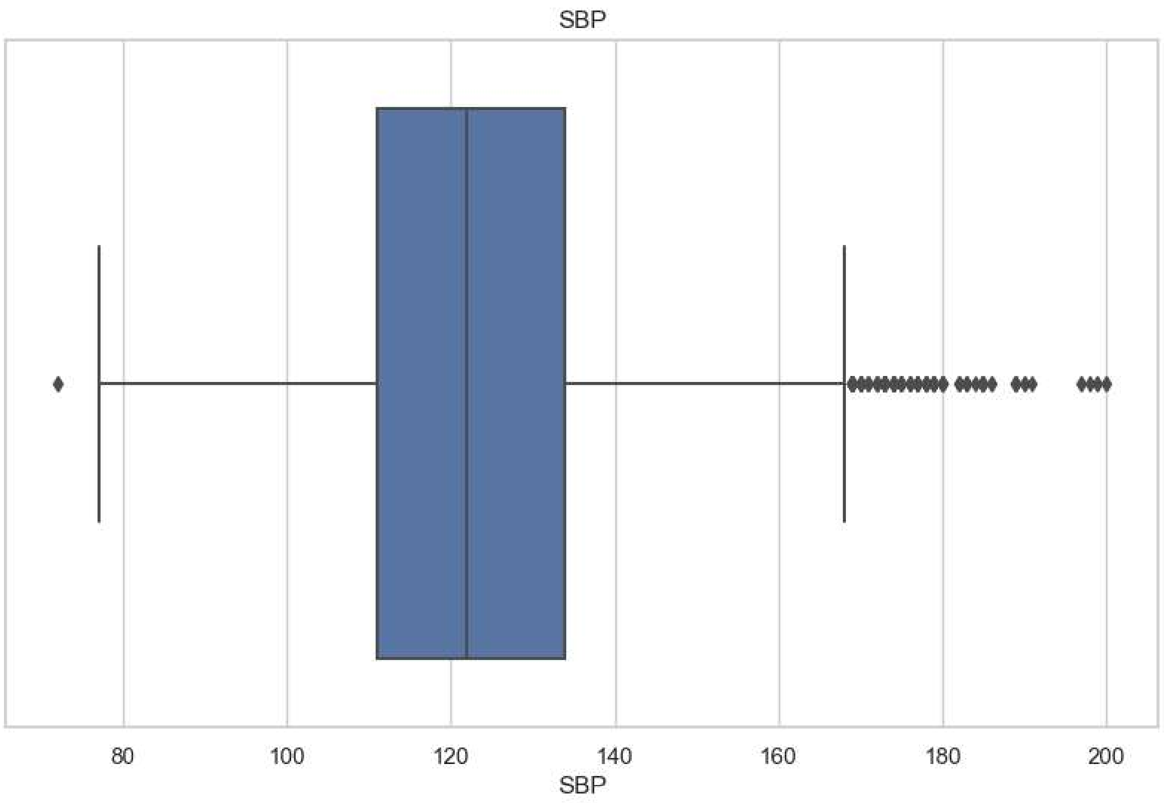
Outliers can significantly skew results. We will employ box plots and IQR methods to visualize and treat outliers, ensuring the model's resilience to anomalous inputs.

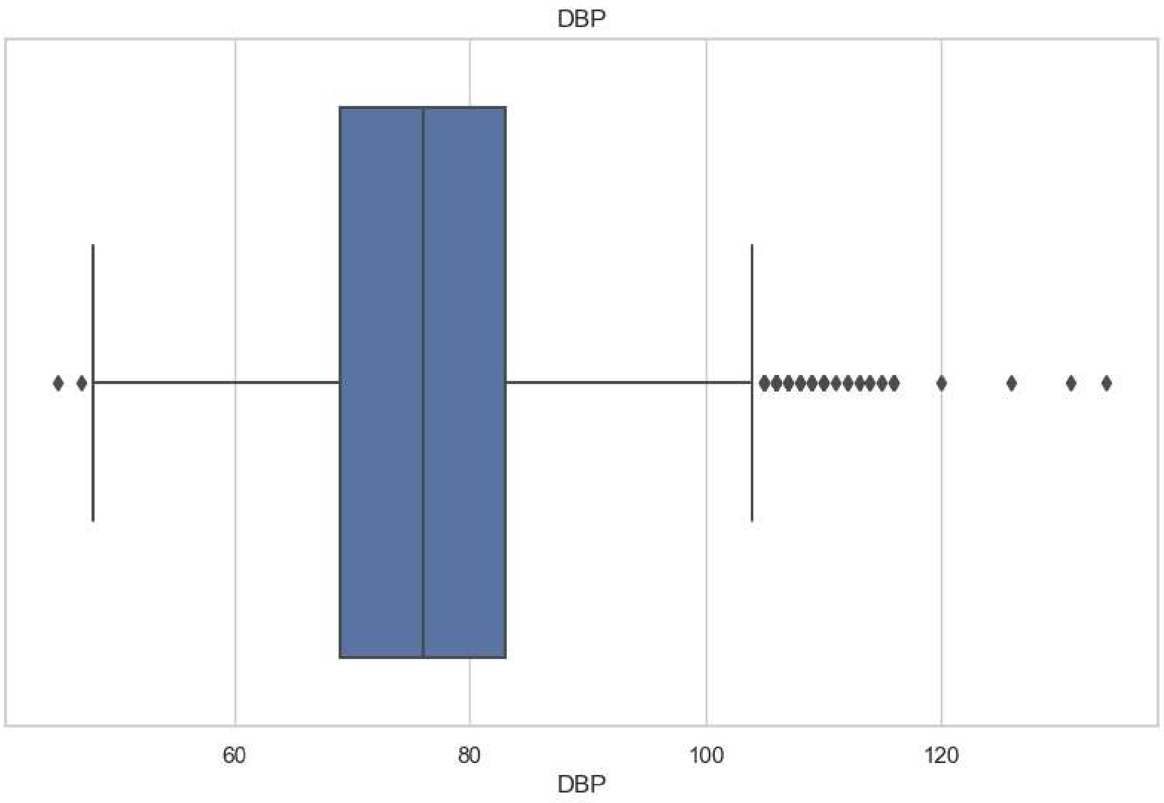
In [26]:
```
# Set the style of the visualization
sns.set(style="whitegrid")

# Loop through each feature in your dataframe
for column in df.columns:
    # Create a figure and a set of subplots
    plt.figure(figsize=(10, 6))

    # Create a boxplot for each feature
    sns.boxplot(x=df[column])

    # Set the title of the plot as the feature name
    plt.title(column)

    # Display the plot
    plt.show()
```

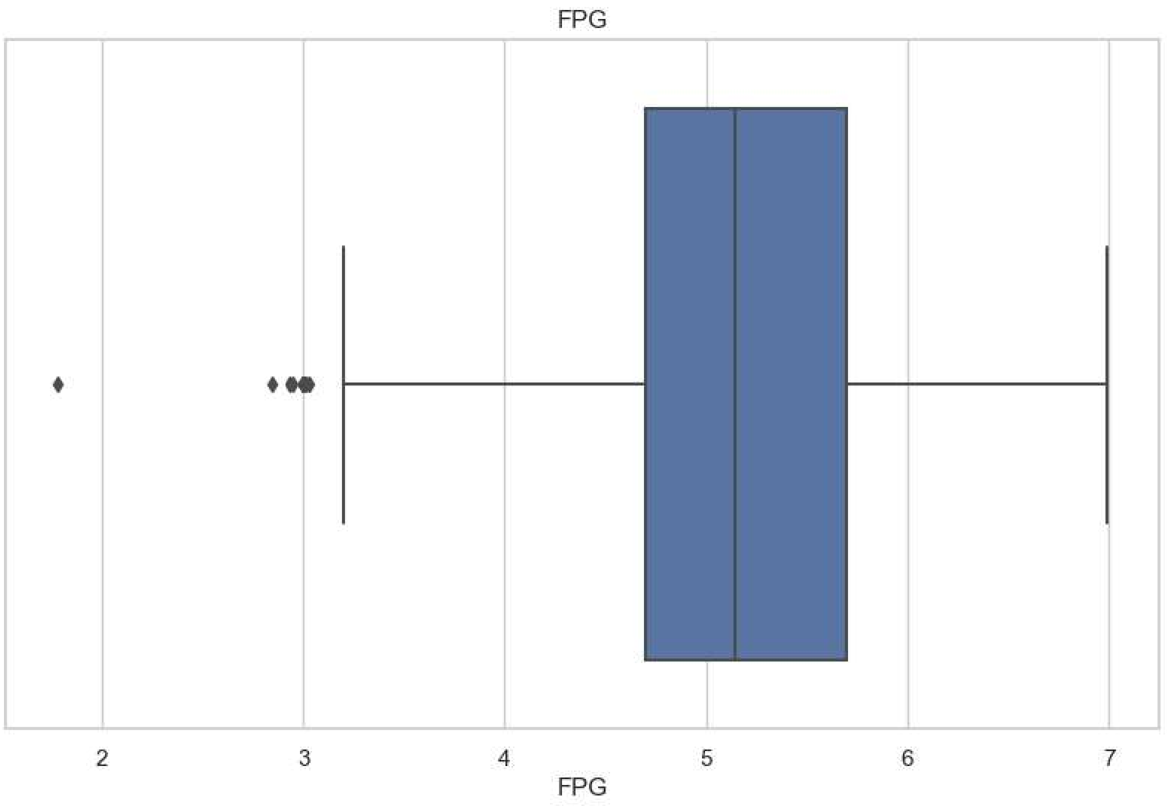


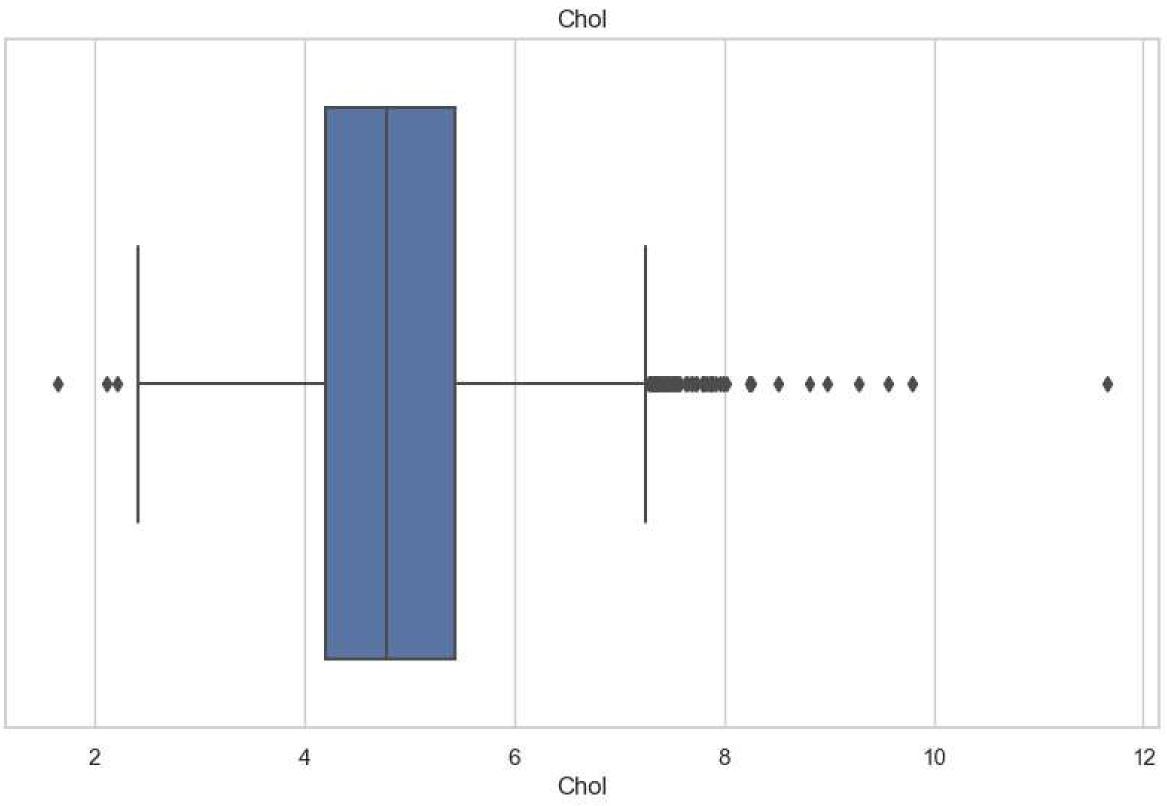

Gender
1.0
1.2
1.4
1.6
1.8
2.0
Gender

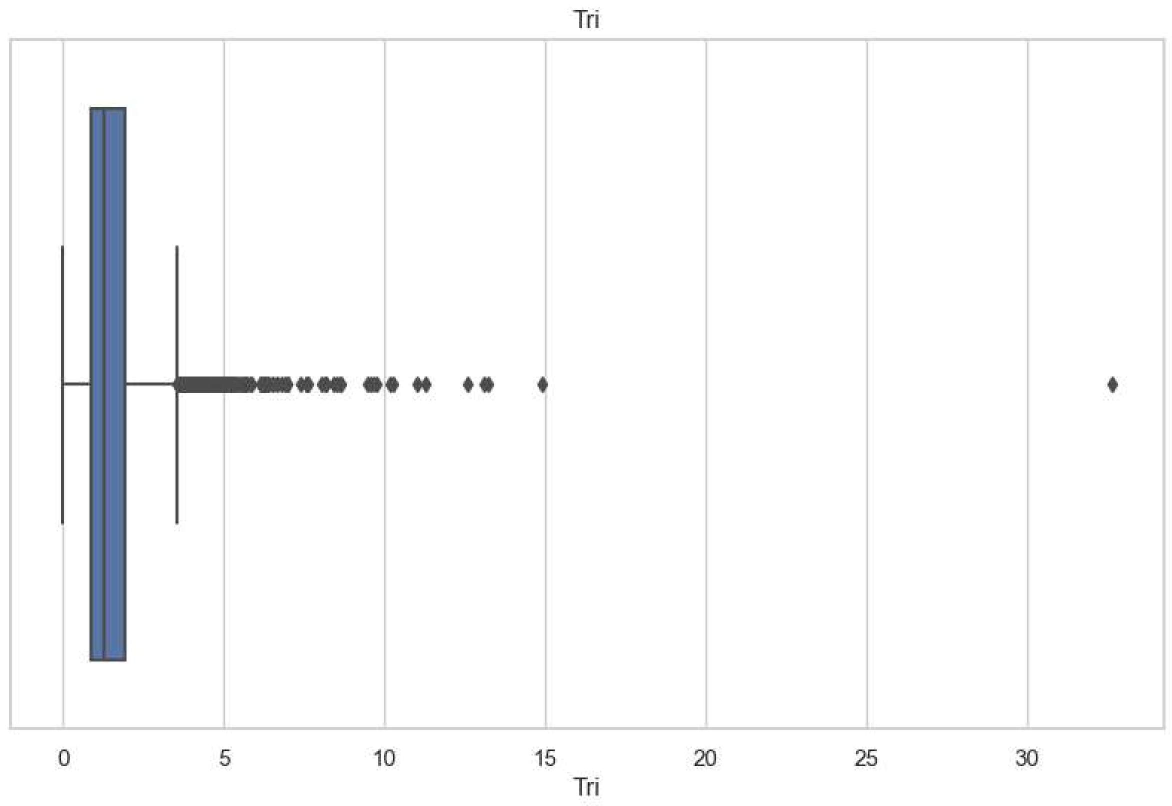

BMI
15
20
25
30
35
40
45
BMI

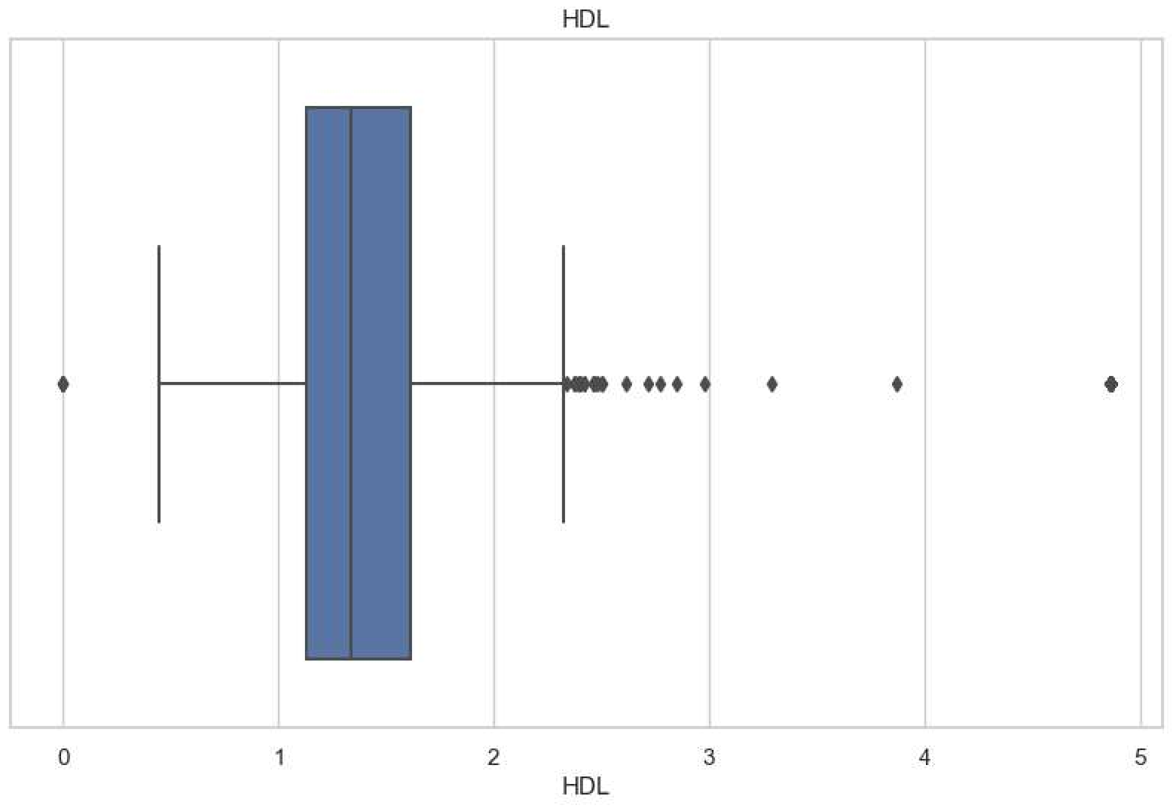

SBP
80
100
120
140
160
180
200
SBP

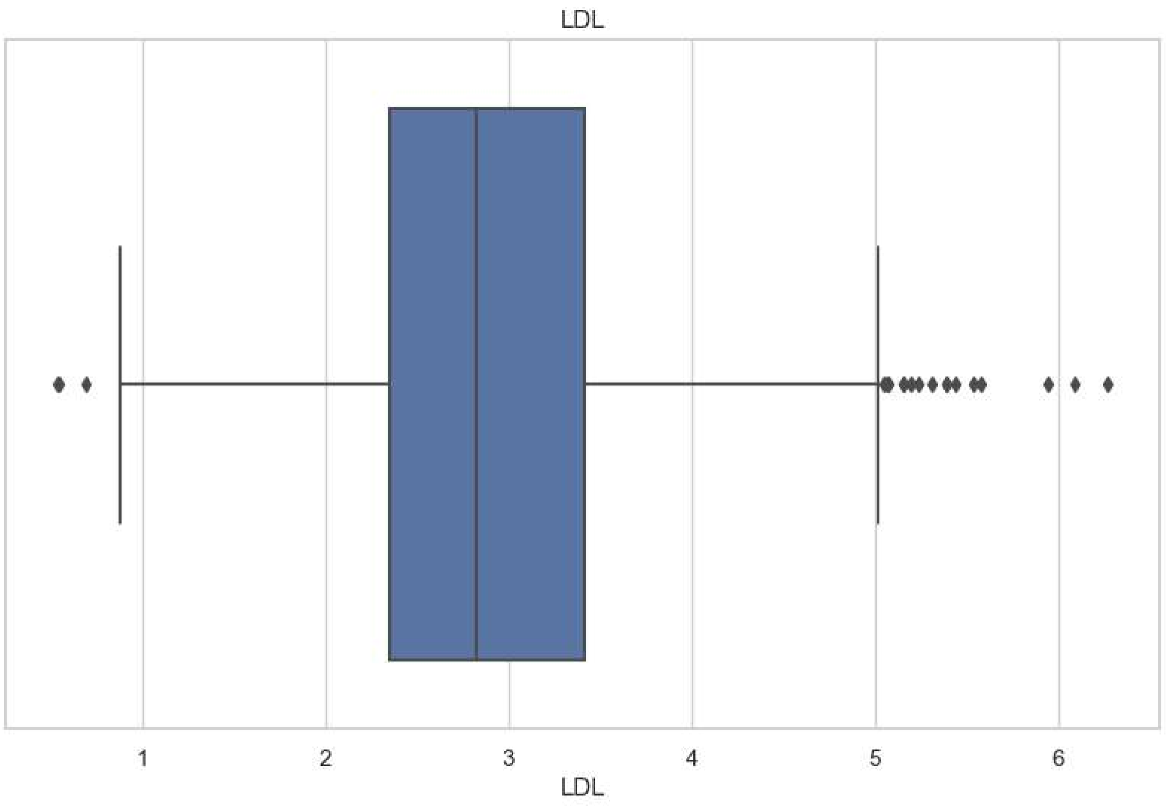

DBP
60
80
100
120
DBP

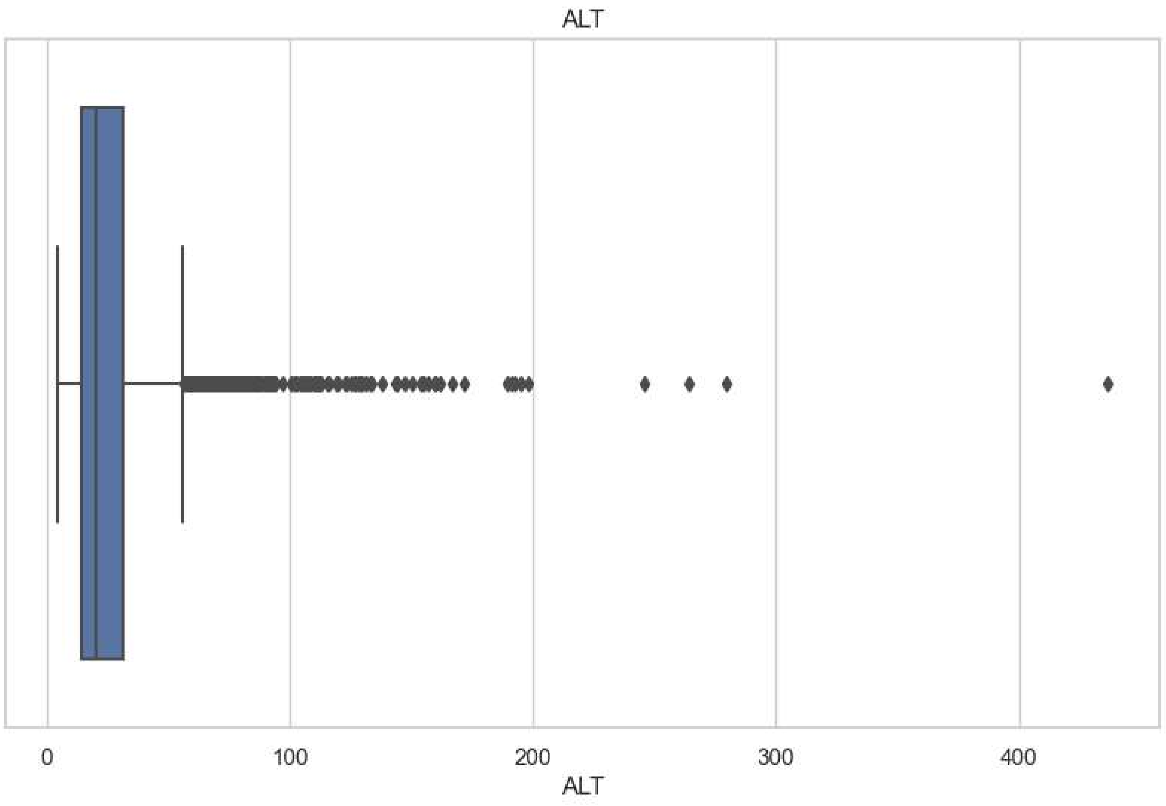

FPG
2
3
4
5
6
7
FPG

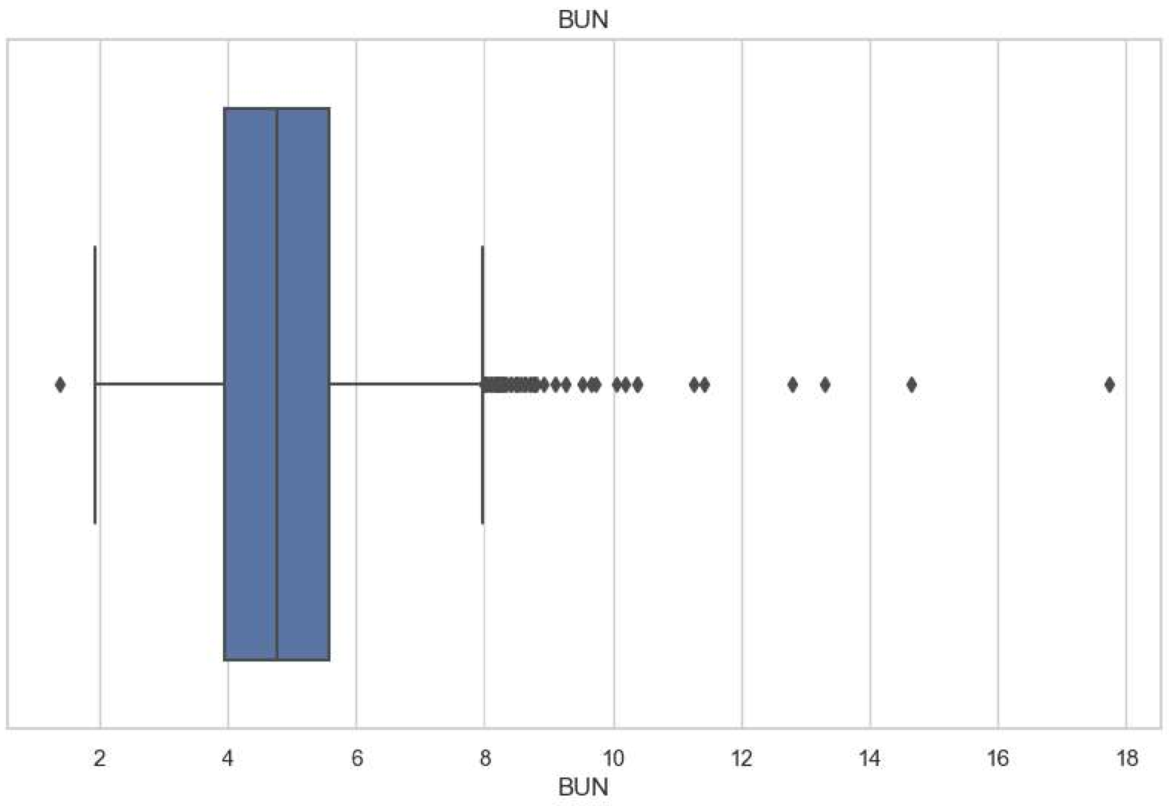

Chol
2
4
6
8
10
12
Chol

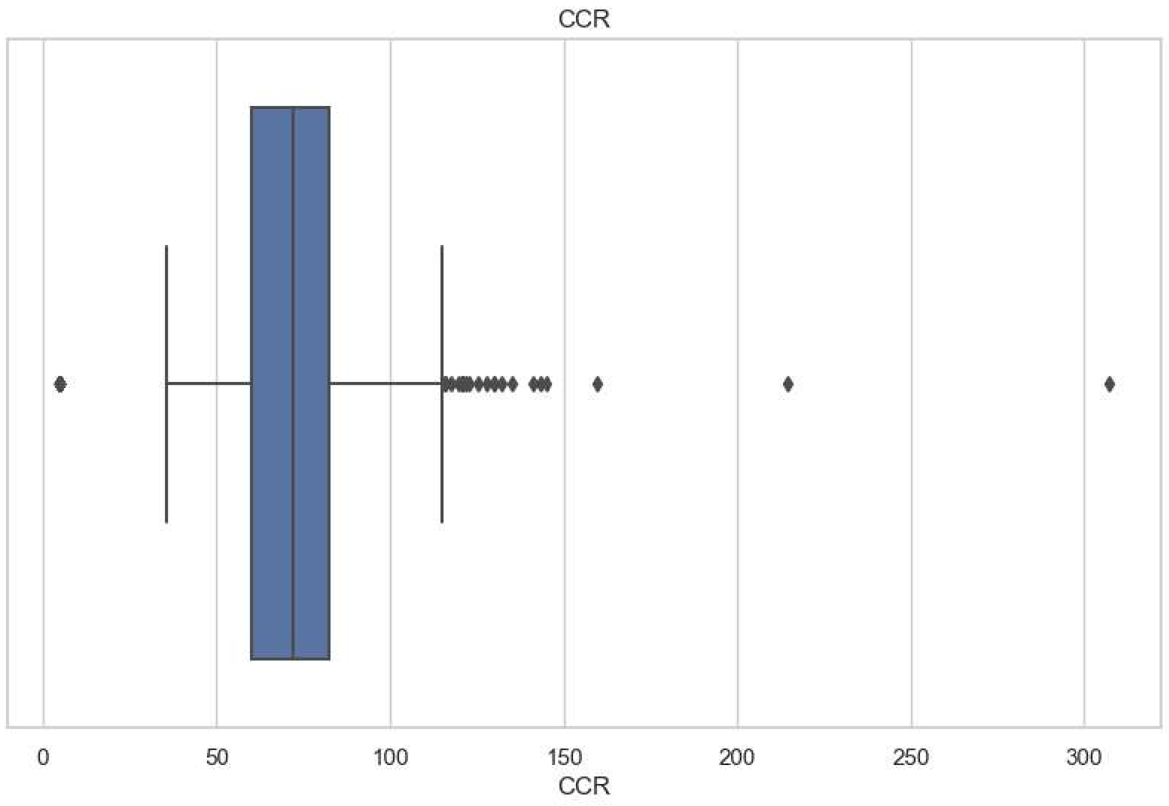

Tri
0
5
10
15
20
25
30
Tri

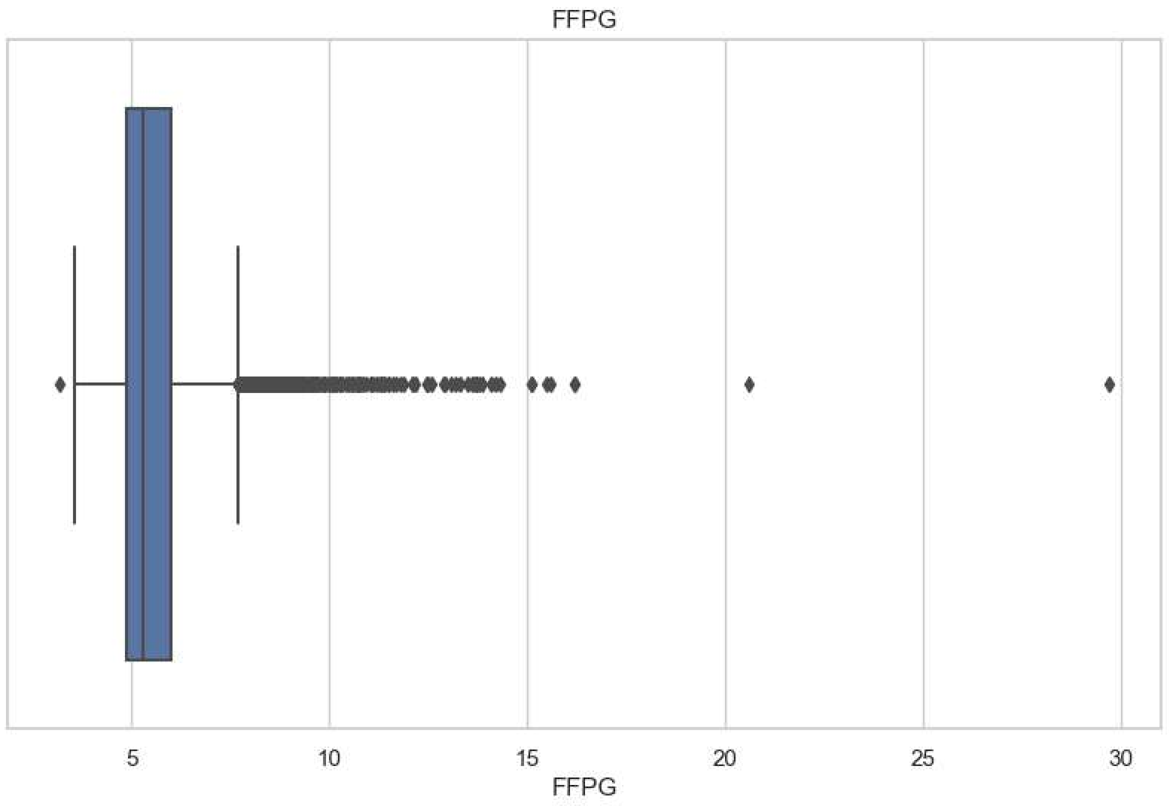

HDL
0
1
2
3
4
5
HDL

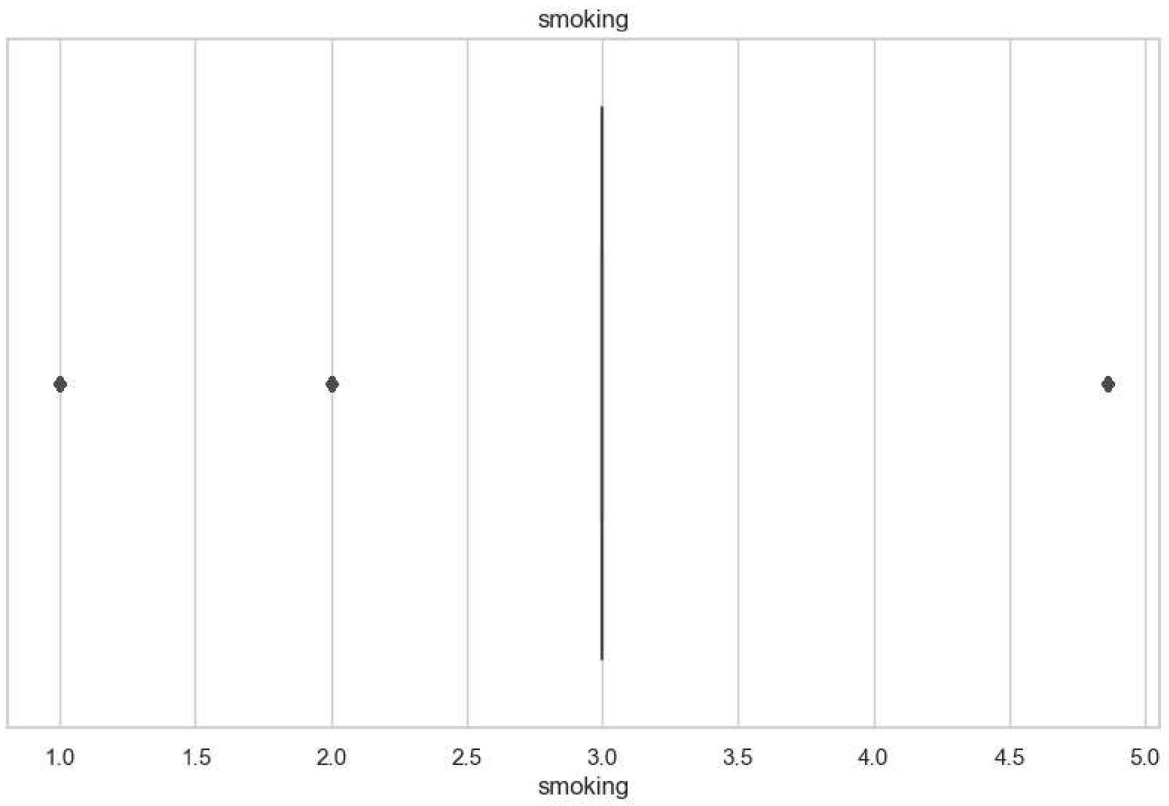

LDL
1
2
3
4
5
6
LDL

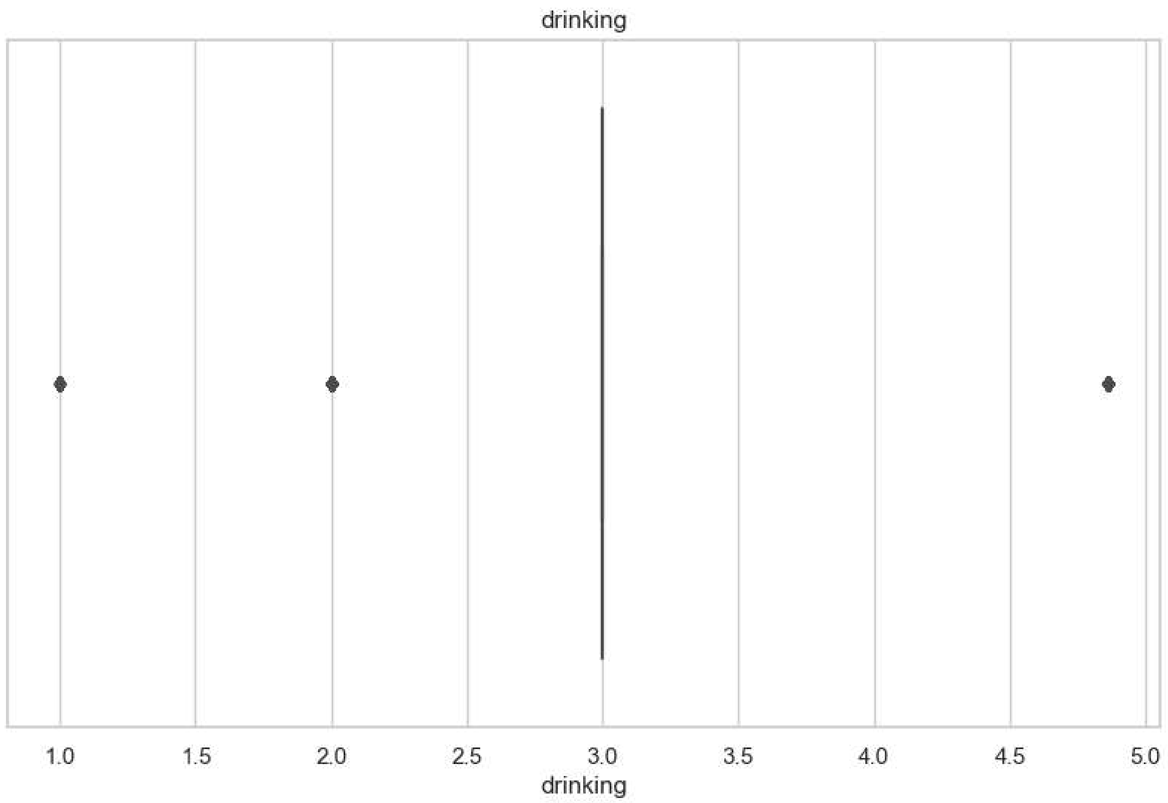

ALT
0
100
200
300
400
ALT

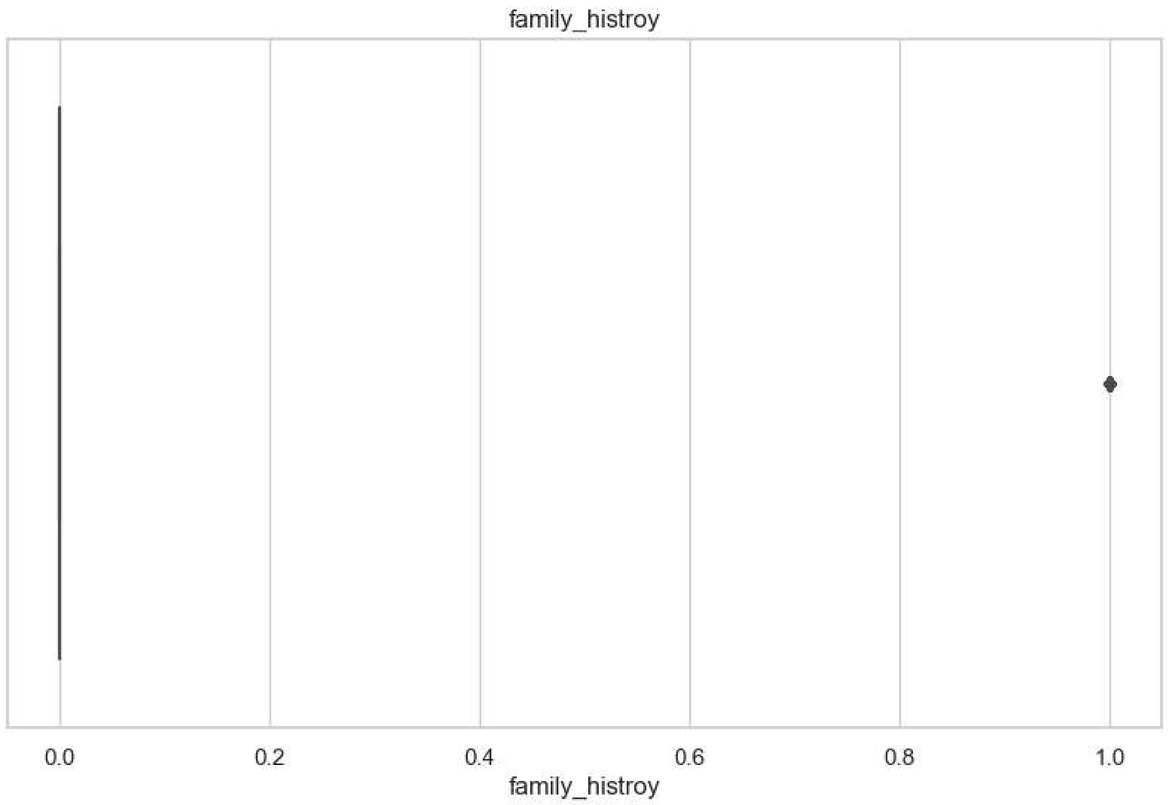

BUN
2
4
6
8
10
12
14
16
18
BUN

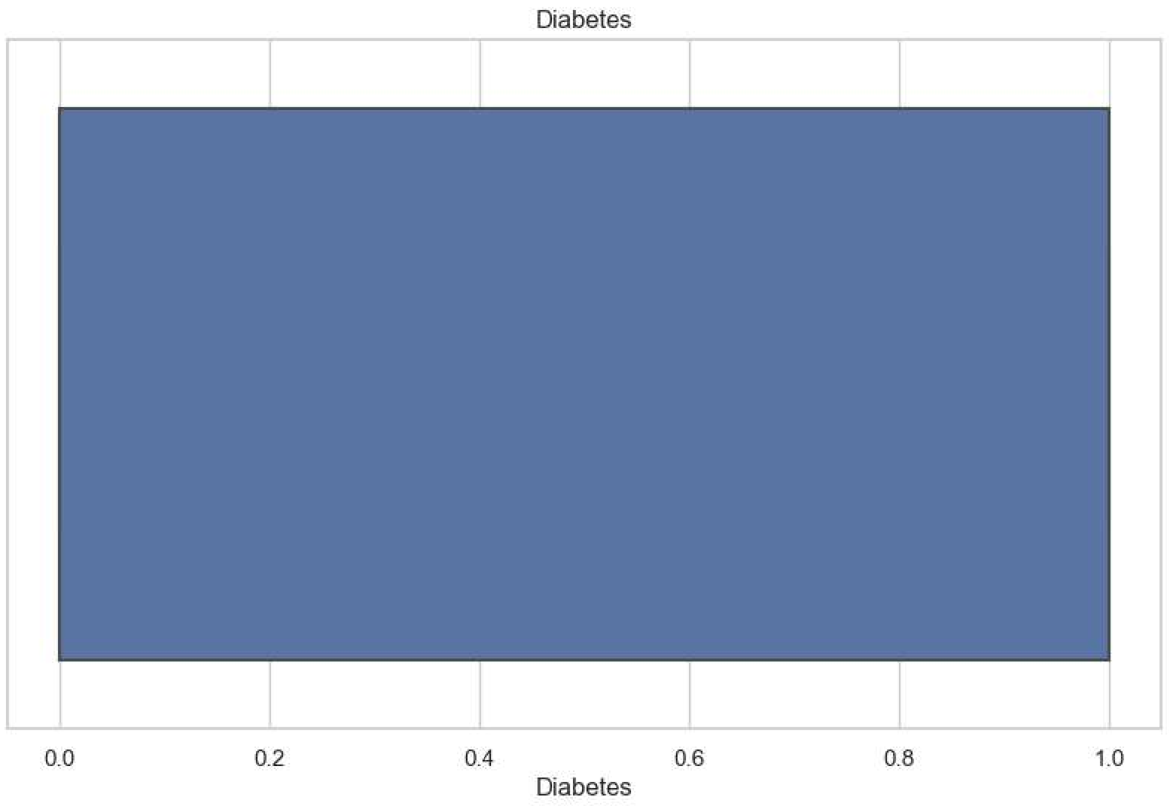

CCR
0
50
100
150
200
250
300
CCR


FFPG
5
10
15
20
25
30
FFPG


smoking
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
smoking


drinking
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
drinking


family_histroy
0.0
0.2
0.4
0.6
0.8
1.0
family_histroy



Our analysis reveals the presence of strong positive outliers in features numbered 3 through 14, with the exception of feature 6, which represents Fasting Plasma Glucose levels. Here, we observe the occurrence of negative outliers. While these outliers, have the potential to introduce variability and influence the overall analysis results, we have opted to retain these within our dataset at this juncture. The decision stems from our cautious approach, ensuring that we do not inadvertently compromise the integrity and comprehensiveness of our data by excluding potentially significant points. We recognize that outliers, while occasionally distortive, can also hold valuable information and provide insights into unique or extreme cases within the studied phenomena.

However, we remain acutely aware of the existence of these outliers and the possible implications they may have on the performance and accuracy of our classifiers. Our approach is adaptive and pragmatic – should we encounter challenges or inconsistencies in the application of our classifiers, we are prepared to reassess their inclusion. Revisiting and potentially adjusting these features by removing outliers would be a considered step to mitigate any distortive impact and enhance the reliability of our model.

Handling Missing Values

The identification and apt handling of missing values is crucial to prevent model degradation. We will discern the pattern of missingness and apply suitable imputation methods or deletion to handle it, if missing values exist.

In [18]:
```
# Check for missing values
print(df.isnull().sum())
```

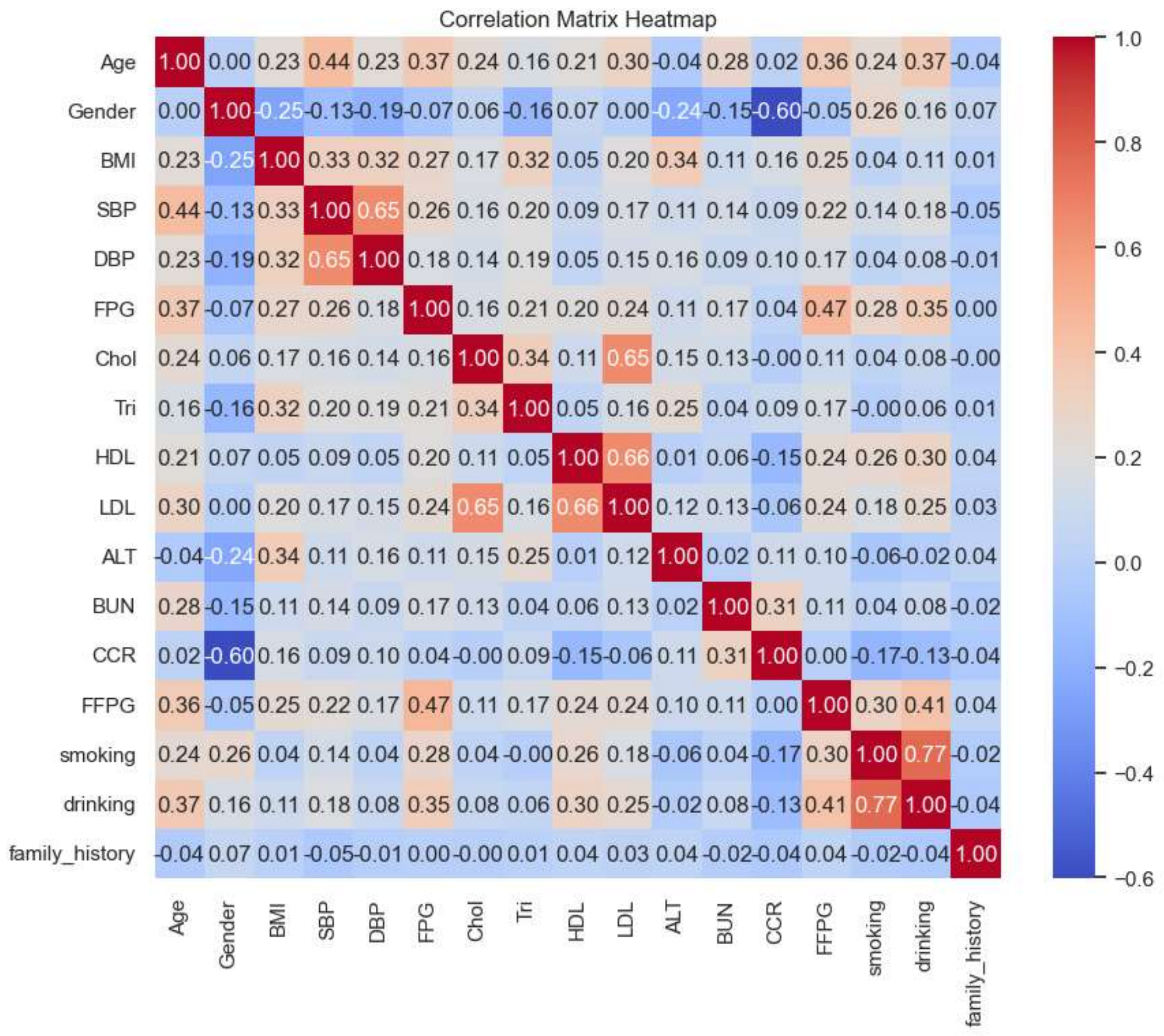
```
Age               0
Gender            0
BMI               0
SBP               0
DBP               0
FPG               0
Chol              0
Tri               0
HDL               0
LDL               0
ALT               0
BUN               0
CCR               0
FFPG              0
smoking           0
drinking          0
family_histroy    0
Diabetes          0
dtype: int64
```

Thankfully, no missing values exist!

Correlation

Understanding the relationships between features and their significance is vital before applying any classifiers. If features are strongly correlated, we can decrease the dimensionality by simpling dropping some of the correlated features. A correlation matrix will unveil the linear relationships between features, guiding feature selection.

In [29]:
```python
correlation_matrix = df.drop('Diabetes', axis = 1).corr()

# Create a heatmap for the correlation matrix
plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f")
plt.title('Correlation Matrix Heatmap')
plt.show()
```



	Age	Gender	BMI	SBP	DBP	FPG	Chol	Tri	HDL	LDL	ALT	BUN	CCR	FFPG	smoking	drinking	family_history
Age	1.00	0.00	0.23	0.44	0.23	0.37	0.24	0.16	0.21	0.30	-0.04	0.28	0.02	0.36	0.24	0.37	-0.04
Gender	0.00	1.00	-0.25	-0.13	-0.19	-0.07	0.06	-0.16	0.07	0.00	-0.24	-0.15	-0.60	-0.05	0.26	0.16	0.07
BMI	0.23	-0.25	1.00	0.33	0.32	0.27	0.17	0.32	0.05	0.20	0.34	0.11	0.16	0.25	0.04	0.11	0.01
SBP	0.44	-0.13	0.33	1.00	0.65	0.26	0.16	0.20	0.09	0.17	0.11	0.14	0.09	0.22	0.14	0.18	-0.05
DBP	0.23	-0.19	0.32	0.65	1.00	0.18	0.14	0.19	0.05	0.15	0.16	0.09	0.10	0.17	0.04	0.08	-0.01
FPG	0.37	-0.07	0.27	0.26	0.18	1.00	0.16	0.21	0.20	0.24	0.11	0.17	0.04	0.47	0.28	0.35	0.00
Chol	0.24	0.06	0.17	0.16	0.14	0.16	1.00	0.34	0.11	0.65	0.15	0.13	-0.00	0.11	0.04	0.08	-0.00
Tri	0.16	-0.16	0.32	0.20	0.19	0.21	0.34	1.00	0.05	0.16	0.25	0.04	0.09	0.17	-0.00	0.06	0.01
HDL	0.21	0.07	0.05	0.09	0.05	0.20	0.11	0.05	1.00	0.66	0.01	0.06	-0.15	0.24	0.26	0.30	0.04
LDL	0.30	0.00	0.20	0.17	0.15	0.24	0.65	0.16	0.66	1.00	0.12	0.13	-0.06	0.24	0.18	0.25	0.03
ALT	-0.04	-0.24	0.34	0.11	0.16	0.11	0.15	0.25	0.01	0.12	1.00	0.02	0.11	0.10	-0.06	-0.02	0.04
BUN	0.28	-0.15	0.11	0.14	0.09	0.17	0.13	0.04	0.06	0.13	0.02	1.00	0.31	0.11	0.04	0.08	-0.02
CCR	0.02	-0.60	0.16	0.09	0.10	0.04	-0.00	0.09	-0.15	-0.06	0.11	0.31	1.00	0.00	-0.17	-0.13	-0.04
FFPG	0.36	-0.05	0.25	0.22	0.17	0.47	0.11	0.17	0.24	0.24	0.10	0.11	0.00	1.00	0.30	0.41	0.04
smoking	0.24	0.26	0.04	0.14	0.04	0.28	0.04	-0.00	0.26	0.18	-0.06	0.04	-0.17	0.30	1.00	0.77	-0.02
drinking	0.37	0.16	0.11	0.18	0.08	0.35	0.08	0.06	0.30	0.25	-0.02	0.08	-0.13	0.41	0.77	1.00	-0.04
family_history	-0.04	0.07	0.01	-0.05	-0.01	0.00	-0.00	0.01	0.04	0.03	0.04	-0.02	-0.04	0.04	-0.02	-0.04	1.00

We observe discernible positive correlations between several sets of variables: smoking and drinking, SBP (Systolic Blood Pressure) and DBP (Diastolic Blood Pressure), and LDL (Low-Density Lipoprotein) and HDL (High-Density Lipoprotein) cholesterol levels. Additionally, our analysis reveals a negative correlation between creatinine clearance and age, suggesting that as age increases, the efficiency of the kidneys in clearing creatinine decreases.

Upon closer inspection of the data, the positive correlation between smoking and drinking suggests a potential lifestyle pattern, while the correlation between SBP and DBP is expected, given their physiological relationship. The correlation between LDL and HDL cholesterol levels is also noteworthy, as they are both crucial indicators of cardiovascular health. Similarly, the negative correlation between creatinine clearance and age is consistent with existing medical knowledge, indicating reduced renal function with advancing age.

Despite the presence of these correlations, a thorough analysis of the strength and impact of each correlation in our study reveals that they are not sufficiently robust to justify omitting any of the features from our analysis. Retaining these features is crucial, as each provides unique insights and contributes to the comprehensiveness and depth of our study. Moreover, the relationships between the features, while evident, are not deterministic, and removing one could potentially lead to a loss of valuable information and reduce the reliability and validity of our findings.

In conclusion, while we acknowledge the existence of these correlations, we advocate for the retention of all features in our dataset to ensure a holistic and nuanced understanding of the