Linear Discriminant Analysis

In [32]:
```
lda = LDA(n_components = 2)

X_train_lda = lda.fit_transform(X_train,y_train)
X_test_lda = lda.transform(X_test)

classifier_lda = LogisticRegression(random_state = random_state)
classifier_lda.fit(X_train_lda, y_train)

y_pred_lda = classifier_lda.predict(X_test_lda)

print(classification_report(y_test, y_pred_lda))
```

```
              precision    recall  f1-score   support

           0       1.00      0.97      0.99       105
           1       0.96      0.99      0.97        91
           2       0.94      0.96      0.95        92
           3       0.97      0.96      0.96       112

    accuracy                           0.97       400
   macro avg       0.97      0.97      0.97       400
weighted avg       0.97      0.97      0.97       400
```

Confusion Matrix

In [33]:
```
cm = confusion_matrix(y_test, y_pred_lda)

df1 = pd.DataFrame(columns=["0","1","2","3"], index= ["0","1","2","3"], data= cm )

f,ax = plt.subplots(figsize=(6,6))

sns.heatmap(df1, annot=True,cmap="Greens", fmt= '.0f',ax=ax,linewidths = 5, cbar = False,annot_kws={"size": 16})
plt.xlabel("Predicted Label")
plt.xticks(size = 12)
plt.yticks(size = 12, rotation = 0)
plt.ylabel("True Label")
plt.title("Confusion Matrix", size = 12)
plt.show()
```