Although the only real data science project I did was the titanic's problem while studying the basics (https://www.kaggle.com/branquinhofelipe), I wanted to submit something to this application. After some google search I found the puzzle. This file has the walk through of my solution. I use Python 3 in jupyter notebook with the following libraries:

Numpy (1.16.1)
Pandas (0.23.4)
Matplotlib (3.0.0)
Seaborn (0.9.0)
sklearn (0.20.2)

1. Read the train dataset.
2. Check features types and missing values.
3. Remove columns and rows that wouldn't help the model.
4. Transform object datas to categorical numbers.
5. Fill few missing float datas with column mean value.
6. Process datetime features (last_payment and end_last_loan) .
    a. If end_last_loan missing, default = False. Customer contract finished.
    b. If last_payment missing, default = True (only if end_last_loan not null). Customer payment pending.
    c. Otherwise, get time elapsed since column min date. Scaled over column total time elapsed.
7. Split train dataset with 20% test_size (calling X_test).
8. Test a RandomForestClassifier for significant features.
    a. Removed the least relevant to prevent overfitting the model.
9. Scale remaining features with Robust Scaler to avoid outliers.
10. Use the GridSearchCV to find the RandomForestClassifier best estimator.
    a. Parameters tested: n_estimators, max_depth and max_features.
    b. Parameters values returned: 200, 9, 0.75 (Respectively).
    c. Parameters score: 0.91494
11. Evaluate the model with the X_test.
    a. Confusion matrix: [[9775 316]
    [668 1219]]
    b. Classification report:

	precision	recall	f1-score	support
False	0.94	0.97	0.95	10091
True	0.79	0.65	0.71	1887

    c. Score: 0.91785
12. Redo steps 1-6 for the test dataset.
13. Scale the test features with Robust Scaler to avoid outliers.
14. Make the prediction with trained model.

During this project I found that it is important to pay attention to outliers. At first, without scaling the dataset, the model predicted all default as False that would mean some money loss for the lender. Another thing to notice is the high precision and recall for the False default, that occurs because of the unbalanced data (Only 18.85% positive) and implies that customers are more willing to pay the loans.