

# Low Level Design (LLD)

Credit risk assessment using South German credit data

December 31, 2021

Revision number: 1.0

Last date of revision: December 31, 2021

# Contents

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Introduction</b>                     | <b>3</b>  |
| 1.1      | Purpose of LLD . . . . .                | 4         |
| 1.2      | Scope . . . . .                         | 4         |
| <b>2</b> | <b>Architecture</b>                     | <b>5</b>  |
| 2.1      | Architecture diagram . . . . .          | 6         |
| <b>3</b> | <b>Architecture description</b>         | <b>7</b>  |
| 3.1      | Data description . . . . .              | 8         |
| 3.2      | Data insertion into database . . . . .  | 8         |
| 3.3      | Export data from the database . . . . . | 8         |
| 3.4      | Data preprocessing . . . . .            | 8         |
| 3.5      | Model building . . . . .                | 8         |
| 3.6      | Data from a user . . . . .              | 9         |
| 3.7      | Model call . . . . .                    | 9         |
| 3.8      | Deployment . . . . .                    | 9         |
| <b>4</b> | <b>Unit tests</b>                       | <b>10</b> |
| 4.1      | Unit test cases . . . . .               | 11        |

# Chapter 1

## Introduction

## 1.1 Purpose of LLD

[Return to table of contents](#)

LLD provides internal design of actual program code for the credit risk assessment solution. This document describes class diagrams with methods and relations between classes and program specs. LLD is constructed in a way that a programmer can code the program from it.

## 1.2 Scope

[Return to table of contents](#)

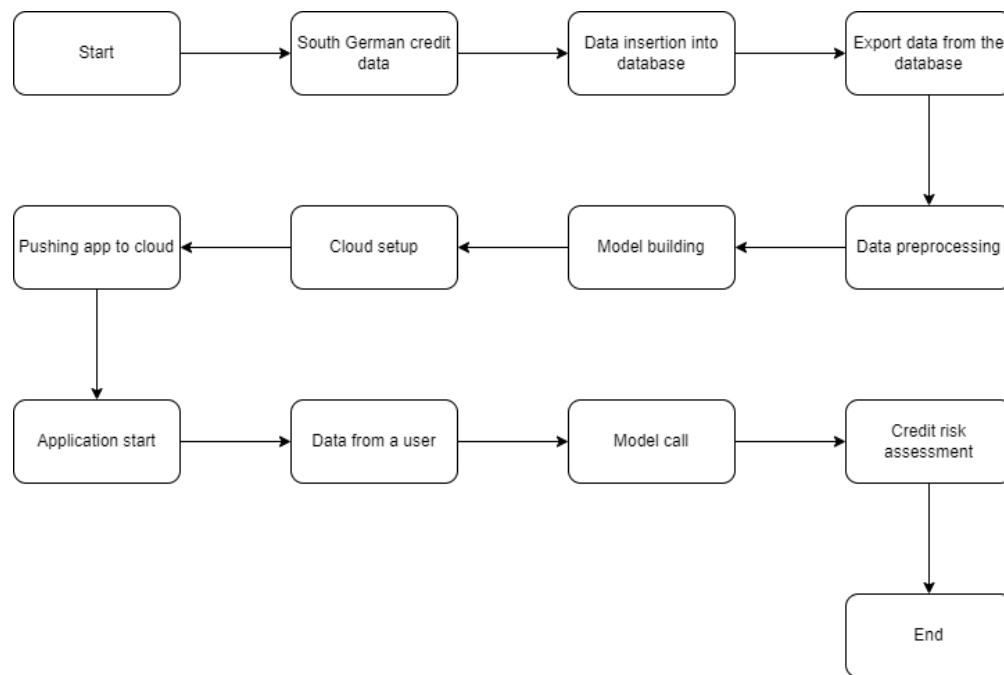
LLD is a component level design process which passes through a step-by-step refinement. The process can be used for designing data structures, required software architecture, source code, and performance algorithms. Data organization may be defined during requirement analysis and then refined during data design work.

## Chapter 2

# Architecture

## 2.1 Architecture diagram

[Return to table of contents](#)



## Chapter 3

# Architecture description

## 3.1 Data description

[Return to table of contents](#)

The data contains 1000 observations for South German bank credit market. Specifically 700 good and 300 bad credits with 20 predictor variables (features). It covers a period from 1973 to 1975. The data comes as a asc file.

## 3.2 Data insertion into database

[Return to table of contents](#)

1. Database creation and connection (if the database exists, open connection to it)
2. Import the data

## 3.3 Export data from the database

[Return to table of contents](#)

The data is exported from the database for data preprocessing and model building.

## 3.4 Data preprocessing

[Return to table of contents](#)

This process includes changing the features names (actual names are in German), creation of new features (continuous variable discretization, hierarchical clustering based on continuous variables, etc.), transforming continuous variables (applying Box-Cox and MinMax transformation), and restructuring the target variable (credit risk) to represent bad credits as 1 and good credits as 0.

## 3.5 Model building

[Return to table of contents](#)

Model building includes construction of 3 models (LogisticRegression with onehot encoded categorical variables, LogisticRegression with weight of evidence features, and a simple DecisionTree) and averaging their outputs to get final result.



## 3.6 Data from a user

[Return to table of contents](#)

At this stage a user (a credit officer) inserts necessary characteristics of an applicant such as age, credit amount, credit duration, and etc.

## 3.7 Model call

[Return to table of contents](#)

The estimated models will be called to assess credit risk.

## 3.8 Deployment

[Return to table of contents](#)

The solution will be deployed to Heroku.

## Chapter 4

## Unit tests

## 4.1 Unit test cases

[Return to table of contents](#)

Testing has been conducted only for a function and a class related to weight of evidence transformation.