**Loan Amortization Calculator**

**Technical Design**

**CMSC 495 6381 Current Trends and Projects in Computer Science (2208)**

**Group Charlie**

James Lisle

Aniebiet Jacob

Mark Tasker

Catherine Wingfield

Sam Shanzhong Yuan

***University of Maryland University College***

Revision: 1.0.2

Date: 09/13/2020

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision** | **Date** | **Name(s) of Contributors** | **Description** |
| 1.0 | 09/13/2020 | Lisle | Original Document |
| 1.0.1 | 09/13/2020 | Jacob | Sequence Diagram |
| 1.0.2 | 09/14/2020 | Wingfield | Axis2 Class Design |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Purpose**

This document presents a technical design of the Loan Amortization Calculator including the rationale for the calculator, feature set, system architecture and class design, technology utilized for implementation, and basic performance estimates.

**Objectives**

This document outlines and details the following factors that are considered to be relevant in the technical design document of the Loan Amortization Calculator:

* Why create a loan amortization calculator
* What are the necessary features
* Proposed Design with supporting classes
* System Architecture
* Technology for implementation
* Performance estimates

**Another Loan Amortization Calculator**

The field is crowded when it comes to online amortization calculators but the main purpose of this project is to provide a platform that demonstrates a common utility that anyone who has every required just a simple tool to perform an amortization calculation can perform with the option of generating the associated amortization schedule describing the loan characteristics they have chosen. This calculator also performs interest calculation which is not common among this type of calculator.

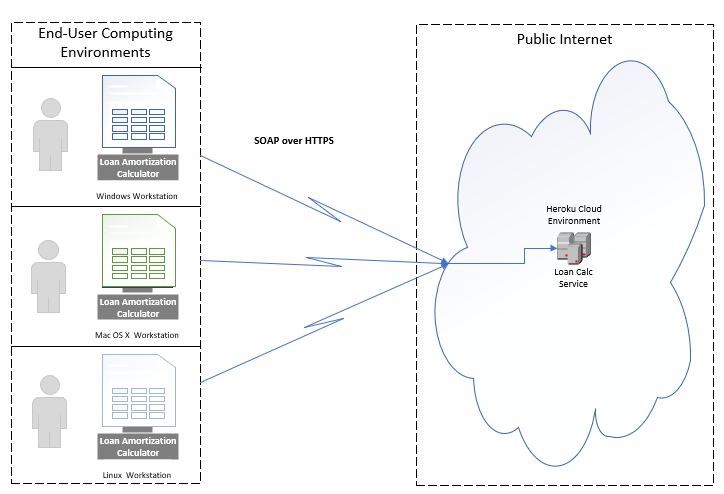
**Basic Feature Set**

* Calculate Loan Amount
* Calculate Loan Payment
* Calculate Loan Rate
* Calculate Loan Term
* Calculate Loan Payoff
* Display Amortization Schedule
* Email Amortization Schedule

**System Design**

The proposed solution for the Loan Amortization Calculator includes a Java Swing Interface that can run on any modern desktop running Windows, Mac OS X, or Linux with Java version 8 or above installed, and an internet connection to a cloud-based service that provides the loan amortization calculation engine.

System Architecture



Class Design

The Loan Amortization Calculator design consists of a Java Swing GUI for selecting the type of amortization calculation desired and a remotely located service running on Tomcat 9.x in the Heroku Cloud to perform the actual calculation. The components communicate securely using the SOAP 1.2 protocol over HTTPS in a synchronous call.

**Client**

The client component implements an MVC design pattern using Java Swing. A high level view of the classes used to implement the GUI are shown in figure 2.

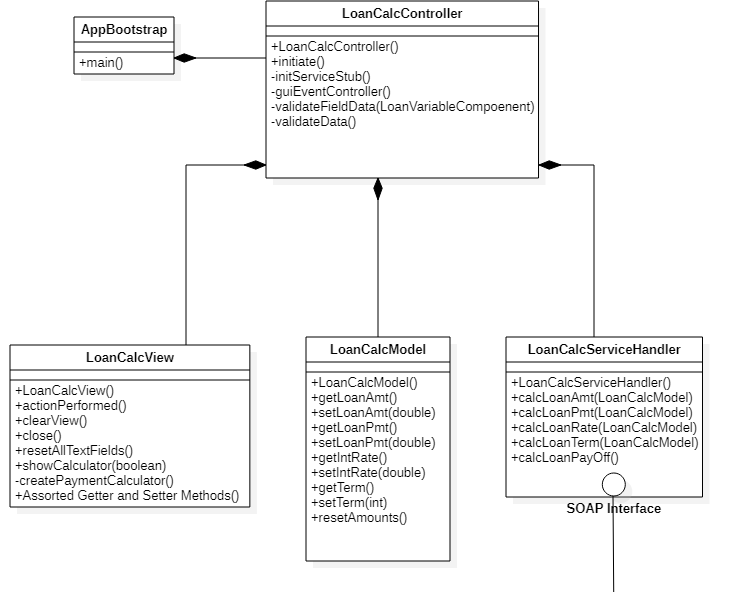


Figure 2 - Loan Amortization Calculator Client Classes

**Service**

The service component consists of a calculation class wrapped with a set of auto-generated classes by Axis2 to marshal/demarshal the SOAP messages and route them to the corresponding operation. A high level view of the classes used to implement the Service are shown in figures 3 and 4a,4b,&4c.

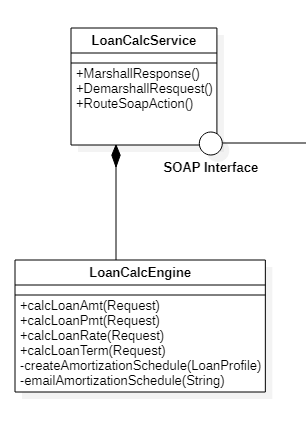
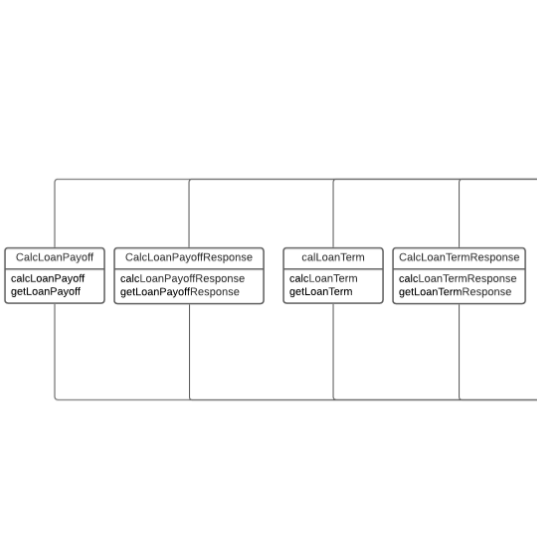
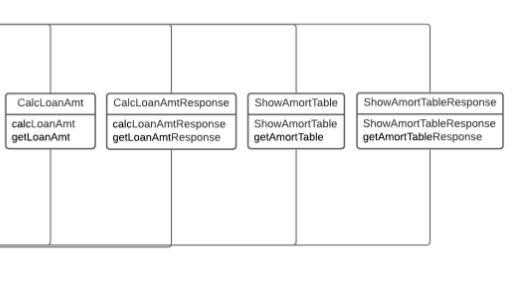


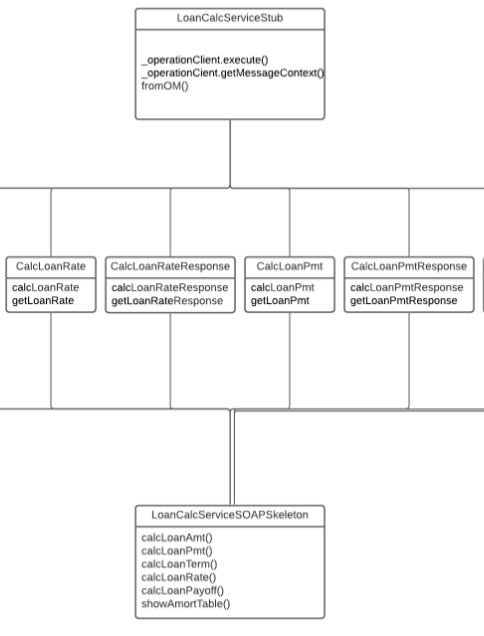
Figure 3 - Loan Amortization Calculator Service Classes



**Figure 4a – Axis2 Request and Response Classes**



**Figure 4b – Axis2 Request and Response Classes**



**Figure 4c – Axis2 Request, Response, Stub, and Skeleton Classes**

**WSDL Specification**

The service is defined by a WSDL specification shown in figure 5. In brief this file specifies the schema of the service, the associated messages, a port type that defines an operation and the associated request and response messages, a binding which defines the ‘style’ and transport of the message which generally are either RPC or Document and HTTPS or SOAP, respectively.

<?xml version=*"1.0"* encoding=*"UTF-8"* standalone=*"no"*?>

<wsdl:definitions xmlns:soap=*"http://schemas.xmlsoap.org/wsdl/soap/"* xmlns:tns=*"http://www.amortizer.org/LoanCalcService/"* xmlns:wsdl=*"http://schemas.xmlsoap.org/wsdl/"* xmlns:xsd=*"http://www.w3.org/2001/XMLSchema"* name=*"LoanCalcService"* targetNamespace=*"http://www.amortizer.org/LoanCalcService/"*>

<wsdl:types>

<xsd:schema targetNamespace=*"http://www.amortizer.org/LoanCalcService/"*>

<xsd:element name=*"CalcLoanAmt"*>

<xsd:complexType>

<xsd:sequence>

<xsd:element name=*"loanPmt"* type=*"xsd:double"*/>

<xsd:element name=*"loanTerm"* type=*"xsd:int"*/>

<xsd:element name=*"loanRate"* type=*"xsd:double"*/>

</xsd:sequence>

</xsd:complexType>

</xsd:element>

<xsd:element name=*"CalcLoanAmtResponse"*>

<xsd:complexType>

<xsd:sequence>

<xsd:element name=*"loanAmt"* type=*"xsd:double"*/>

</xsd:sequence>

</xsd:complexType>

</xsd:element>

<xsd:element name=*"CalcLoanPmt"*>

<xsd:complexType>

<xsd:sequence>

<xsd:element name=*"loanAmt"* type=*"xsd:double"*/>

<xsd:element name=*"loanRate"* type=*"xsd:double"*/>

<xsd:element name=*"loanTerm"* type=*"xsd:int"*/>

</xsd:sequence>

</xsd:complexType>

</xsd:element>

<xsd:element name=*"CalcLoanPmtResponse"*>

<xsd:complexType>

<xsd:sequence>

<xsd:element name=*"loanPmt"* type=*"xsd:double"*/>

</xsd:sequence>

</xsd:complexType>

</xsd:element>

<xsd:element name=*"CalcLoanTerm"*>

<xsd:complexType>

<xsd:sequence>

<xsd:element name=*"loanAmt"* type=*"xsd:double"*/>

<xsd:element name=*"loanRate"* type=*"xsd:double"*/>

<xsd:element name=*"loanPmt"* type=*"xsd:double"*/>

</xsd:sequence>

</xsd:complexType>

</xsd:element>

<xsd:element name=*"CalcLoanTermResponse"*>

<xsd:complexType>

<xsd:sequence>

<xsd:element name=*"loanTerm"* type=*"xsd:int"*/>

</xsd:sequence>

</xsd:complexType>

</xsd:element>

<xsd:element name=*"CalcLoanRate"*>

<xsd:complexType>

<xsd:sequence>

<xsd:element name=*"loanAmt"* type=*"xsd:double"*/>

<xsd:element name=*"loanPmt"* type=*"xsd:double"*/>

<xsd:element name=*"loanTerm"* type=*"xsd:int"*/>

</xsd:sequence>

</xsd:complexType>

</xsd:element>

<xsd:element name=*"CalcLoanRateResponse"*>

<xsd:complexType>

<xsd:sequence>

<xsd:element name=*"loanRate"* type=*"xsd:double"*/>

</xsd:sequence>

</xsd:complexType>

</xsd:element>

<xsd:element name=*"CalcLoanPayoff"*>

<xsd:complexType>

<xsd:sequence>

<xsd:element name=*"loanAmt"* type=*"xsd:double"*/>

<xsd:element name=*"loanPmt"* type=*"xsd:double"*/>

<xsd:element name=*"loanRate"* type=*"xsd:double"*/>

<xsd:element name=*"loanTerm"* type=*"xsd:int"*/>

</xsd:sequence>

</xsd:complexType>

</xsd:element>

<xsd:element name=*"CalcLoanPayoffResponse"*>

<xsd:complexType>

<xsd:sequence>

<xsd:element name=*"loanPayoff"* type=*"xsd:double"*/>

</xsd:sequence>

</xsd:complexType>

</xsd:element>

<xsd:element name=*"ShowAmortTable"*>

<xsd:complexType>

<xsd:sequence>

<xsd:element name=*"loanAmt"* type=*"xsd:double"*/>

<xsd:element name=*"loanPmt"* type=*"xsd:double"*/>

<xsd:element name=*"loanRate"* type=*"xsd:double"*/>

<xsd:element name=*"loanTerm"* type=*"xsd:int"*/>

</xsd:sequence>

</xsd:complexType>

</xsd:element>

<xsd:element name=*"ShowAmortTableResponse"*>

<xsd:complexType>

<xsd:sequence>

<xsd:element name=*"out"* type=*"xsd:string"*/>

</xsd:sequence>

</xsd:complexType>

</xsd:element>

</xsd:schema>

</wsdl:types>

<wsdl:message name=*"CalcLoanAmtRequest"*>

<wsdl:part element=*"tns:CalcLoanAmt"* name=*"parameters"*/>

</wsdl:message>

<wsdl:message name=*"CalcLoanAmtResponse"*>

<wsdl:part element=*"tns:CalcLoanAmtResponse"* name=*"parameters"*/>

</wsdl:message>

<wsdl:message name=*"CalcLoanPmtRequest"*>

<wsdl:part name=*"parameters"* element=*"tns:CalcLoanPmt"*/>

</wsdl:message>

<wsdl:message name=*"CalcLoanPmtResponse"*>

<wsdl:part name=*"parameters"* element=*"tns:CalcLoanPmtResponse"*/>

</wsdl:message>

<wsdl:message name=*"CalcLoanTermRequest"*>

<wsdl:part name=*"parameters"* element=*"tns:CalcLoanTerm"*/>

</wsdl:message>

<wsdl:message name=*"CalcLoanTermResponse"*>

<wsdl:part name=*"parameters"* element=*"tns:CalcLoanTermResponse"*/>

</wsdl:message>

<wsdl:message name=*"CalcLoanRateRequest"*>

<wsdl:part name=*"parameters"* element=*"tns:CalcLoanRate"*/>

</wsdl:message>

<wsdl:message name=*"CalcLoanRateResponse"*>

<wsdl:part name=*"parameters"* element=*"tns:CalcLoanRateResponse"*/>

</wsdl:message>

<wsdl:message name=*"CalcLoanPayoffRequest"*>

<wsdl:part name=*"parameters"* element=*"tns:CalcLoanPayoff"*/>

</wsdl:message>

<wsdl:message name=*"CalcLoanPayoffResponse"*>

<wsdl:part name=*"parameters"* element=*"tns:CalcLoanPayoffResponse"*/>

</wsdl:message>

<wsdl:message name=*"ShowAmortTableRequest"*>

<wsdl:part name=*"parameters"* element=*"tns:ShowAmortTable"*/>

</wsdl:message>

<wsdl:message name=*"ShowAmortTableResponse"*>

<wsdl:part name=*"parameters"* element=*"tns:ShowAmortTableResponse"*/>

</wsdl:message>

<wsdl:portType name=*"LoanCalcService"*>

<wsdl:operation name=*"CalcLoanAmt"*>

<wsdl:input message=*"tns:CalcLoanAmtRequest"*/>

<wsdl:output message=*"tns:CalcLoanAmtResponse"*/>

</wsdl:operation>

<wsdl:operation name=*"CalcLoanPmt"*>

<wsdl:input message=*"tns:CalcLoanPmtRequest"*/>

<wsdl:output message=*"tns:CalcLoanPmtResponse"*/>

</wsdl:operation>

<wsdl:operation name=*"CalcLoanTerm"*>

<wsdl:input message=*"tns:CalcLoanTermRequest"*/>

<wsdl:output message=*"tns:CalcLoanTermResponse"*/>

</wsdl:operation>

<wsdl:operation name=*"CalcLoanRate"*>

<wsdl:input message=*"tns:CalcLoanRateRequest"*/>

<wsdl:output message=*"tns:CalcLoanRateResponse"*/>

</wsdl:operation>

<wsdl:operation name=*"CalcLoanPayoff"*>

<wsdl:input message=*"tns:CalcLoanPayoffRequest"*/>

<wsdl:output message=*"tns:CalcLoanPayoffResponse"*/>

</wsdl:operation>

<wsdl:operation name=*"ShowAmortTable"*>

<wsdl:input message=*"tns:ShowAmortTableRequest"*/>

<wsdl:output message=*"tns:ShowAmortTableResponse"*/>

</wsdl:operation>

</wsdl:portType>

<wsdl:binding name=*"LoanCalcServiceSOAP"* type=*"tns:LoanCalcService"*>

<soap:binding style=*"document"* transport=*"http://schemas.xmlsoap.org/soap/http"*/>

<wsdl:operation name=*"CalcLoanAmt"*>

<soap:operation soapAction=*"http://www.amortizer.org/LoanCalcService/CalcLoanAmt"*/>

<wsdl:input>

<soap:body use=*"literal"*/>

</wsdl:input>

<wsdl:output>

<soap:body use=*"literal"*/>

</wsdl:output>

</wsdl:operation>

<wsdl:operation name=*"CalcLoanPmt"*>

<soap:operation soapAction=*"http://www.amortizer.org/LoanCalcService/CalcLoanPmt"*/>

<wsdl:input>

<soap:body use=*"literal"*/>

</wsdl:input>

<wsdl:output>

<soap:body use=*"literal"*/>

</wsdl:output>

</wsdl:operation>

<wsdl:operation name=*"CalcLoanTerm"*>

<soap:operation soapAction=*"http://www.amortizer.org/LoanCalcService/CalcLoanTerm"*/>

<wsdl:input>

<soap:body use=*"literal"*/>

</wsdl:input>

<wsdl:output>

<soap:body use=*"literal"*/>

</wsdl:output>

</wsdl:operation>

<wsdl:operation name=*"CalcLoanRate"*>

<soap:operation soapAction=*"http://www.amortizer.org/LoanCalcService/CalcLoanRate"*/>

<wsdl:input>

<soap:body use=*"literal"*/>

</wsdl:input>

<wsdl:output>

<soap:body use=*"literal"*/>

</wsdl:output>

</wsdl:operation>

<wsdl:operation name=*"CalcLoanPayoff"*>

<soap:operation soapAction=*"http://www.amortizer.org/LoanCalcService/CalcLoanPayoff"*/>

<wsdl:input>

<soap:body use=*"literal"*/>

</wsdl:input>

<wsdl:output>

<soap:body use=*"literal"*/>

</wsdl:output>

</wsdl:operation>

<wsdl:operation name=*"ShowAmortTable"*>

<soap:operation soapAction=*"http://www.amortizer.org/LoanCalcService/ShowAmortTable"*/>

<wsdl:input>

<soap:body use=*"literal"*/>

</wsdl:input>

<wsdl:output>

<soap:body use=*"literal"*/>

</wsdl:output>

</wsdl:operation>

</wsdl:binding>

<wsdl:service name=*"LoanCalcService"*>

<wsdl:port binding=*"tns:LoanCalcServiceSOAP"* name=*"LoanCalcServiceSOAP"*>

<soap:address location=*"http://www.amortizer.org/"*/>

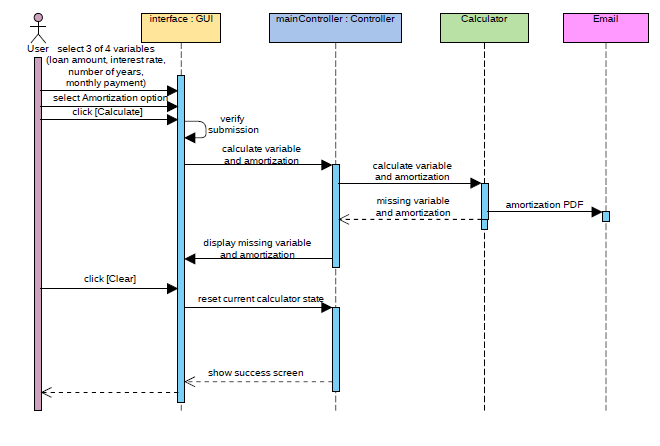
</wsdl:port>

</wsdl:service>

</wsdl:definitions>

**Figure 5 - WSDL specification for the Loan Amortization Calculator**

**Sequence Diagram**



**Figure 6 – Loan Amortization Calculation Sequence (client to service and back)**

**Business Logic**

Loan Payment Calculation:

Loan Amount Calculation:

Loan Term Calculation:

where

* A = loan payment per term
* P = loan principal
* r = interest rate per term
* n = total number of payments

Loan Rate Calculation:

**double** maxRate = 100;

**double** minRate = 0;

**double** midRate = 0;

**double** guessLoanAmt = 0;

**double** monthlyRate = 0;

While (minRate < maxRate – DELTA) {

midRate = (minRate + maxRate)/2

monthlyRate = midrate / (12\*100)

loanAmtGuess = calcLoanAmt(…)

if ( loanAmtGuess < loanAmt)

maxRate = midRate;

else

minRate = midRate;

}

this.rate = midRate

**Data Model**

There is no persistent data model used in this application.

**Performance**

The Loan Amortization Calculator is expected to perform in a sub-second timeframe

**Risk**

The biggest risk to this application is that the service is unavailable.

**Assumptions**

The end user will be able to set the application up with a valid JRE and an internet connection will be available.

**Core Test Cases**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Loan Amount** | **Loan Payment** | **Loan Rate** | **Loan Term** | **Data**  **Inputs** | **Expected**  **Results** | **Actual**  **Results** |
| **Loan amount** | **P** | **1250.00** | **5.375%** | **30 yrs** |  |  |  |
| **Loan payment** | **45000** | **A** | **4.75%** | **5 yrs** |  |  |  |
| **Loan Rate** | **6500** | **600** | **R** | **1 yr** |  |  |  |
| **Loan Term** | **24950** | **699.50** | **7.25%** | **n** |  |  |  |