Table of Contents

Software Design Document (SDD)

1. Introduction

- **Project Overview:**
  Build a REST API that meets CalHFA's requirements while building upon their vision through the addition of a website that contains:

    - Testing space for database requests
    - View of the requested loan information
    - Link directly to aforementioned data in JSON
    - API documentation

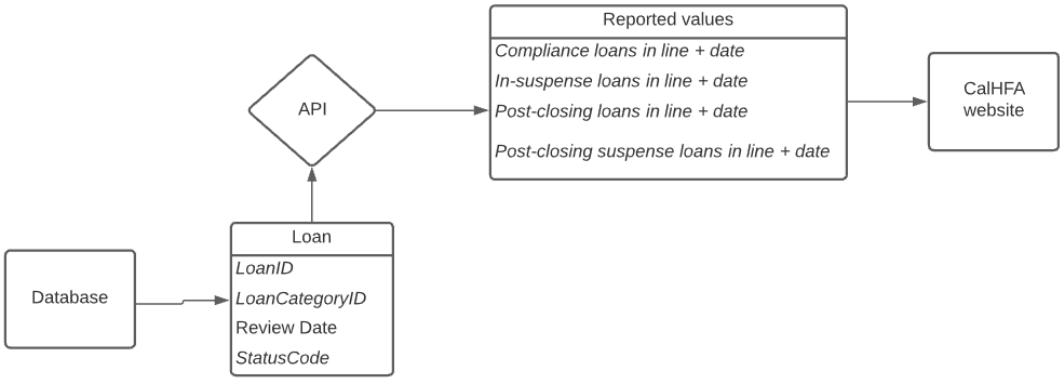
- **Major goals of the system**:
  The ultimate focus of this project was to automate a process that the CalHFA marketing team was formerly doing manually, ensuring that their time spent is being optimized.

2. Functional Requirements

- Get the status codes that end in 10 and 22.
- Refine further by searching for loans that have the description: "Purchase", "Compliance" or "Suspense."
- Count the number of times each status code appears, reference their loan category ID, and log them in their appropriate positions.
- Archive the oldest date of each status code and log them in their appropriate positions.
- Finally, display the accurate loan counts and dates.

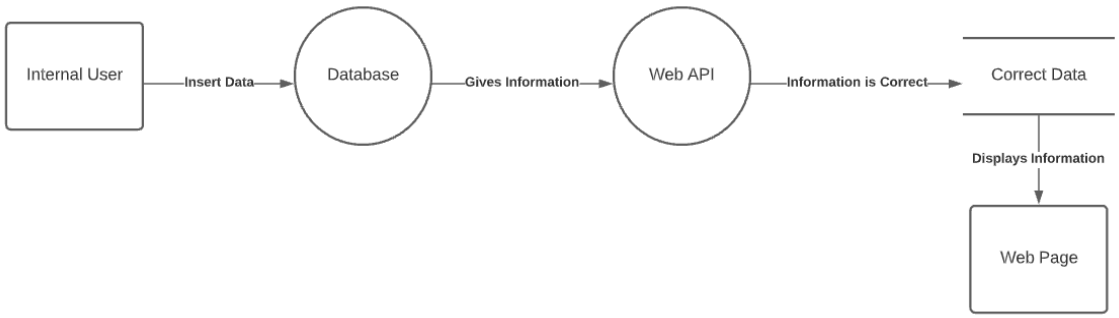
3. Data Design

- The Data Design describes structures that reside within the software. Attributes and relationships between data objects dictate the choice of data structures



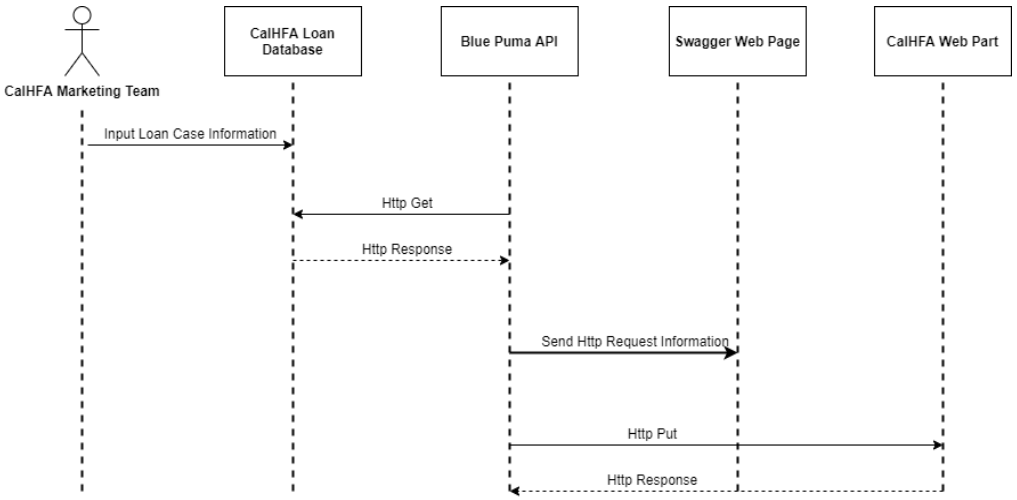
- *Database*: The clients will provide the tables with the correct and up to date information
- *Loan*: Those are the columns needed
- *API*: sort through the data and organize to four different set of values
- *Reported values:* these are the four set of values that will be displayed in the user interface

4. Architectural Design

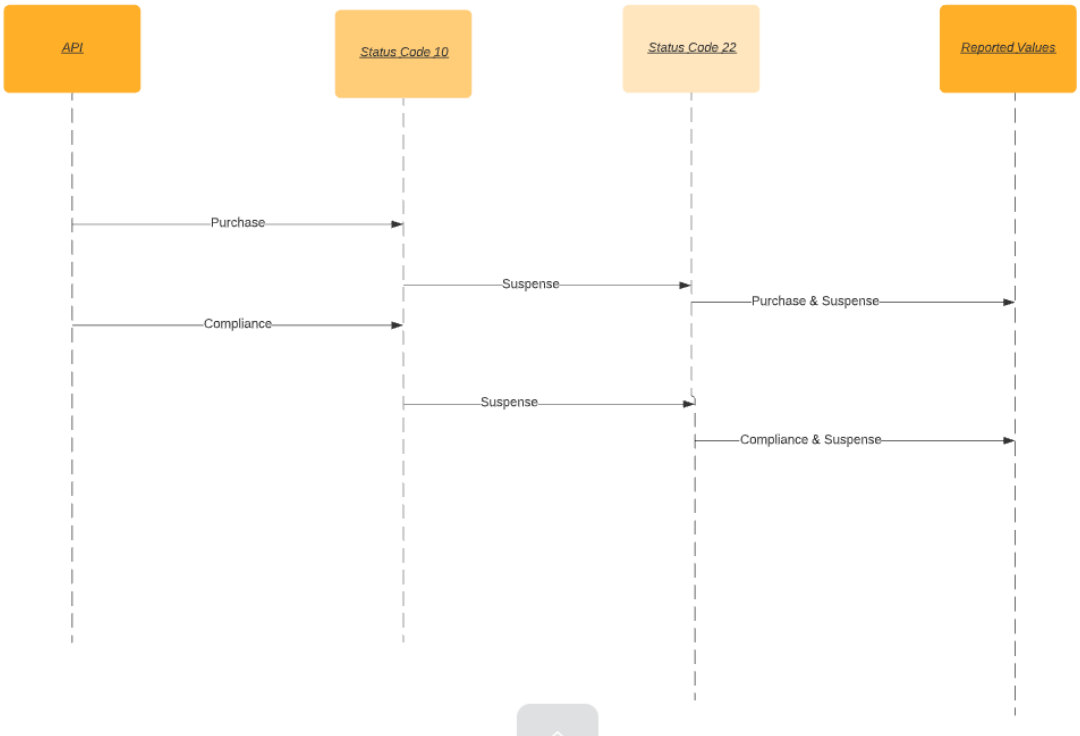


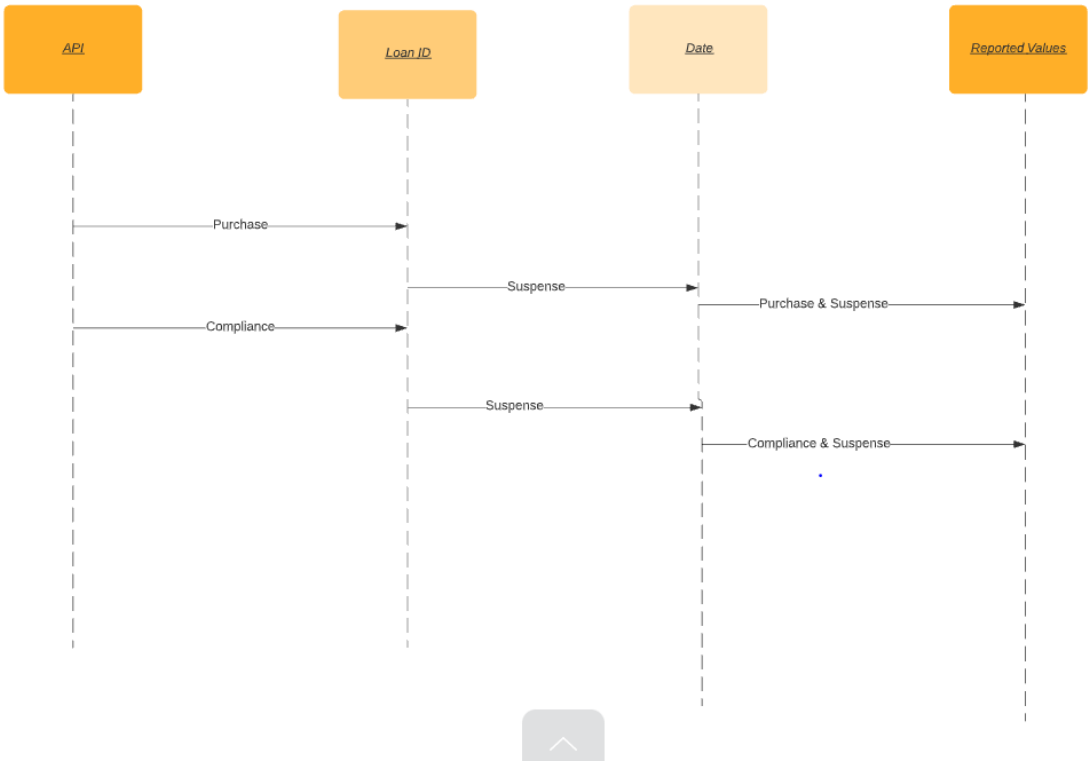
5. Detailed Design

- Create a set of sequence diagrams for the most critical Use Cases



Francia Garcia | December 17, 2021

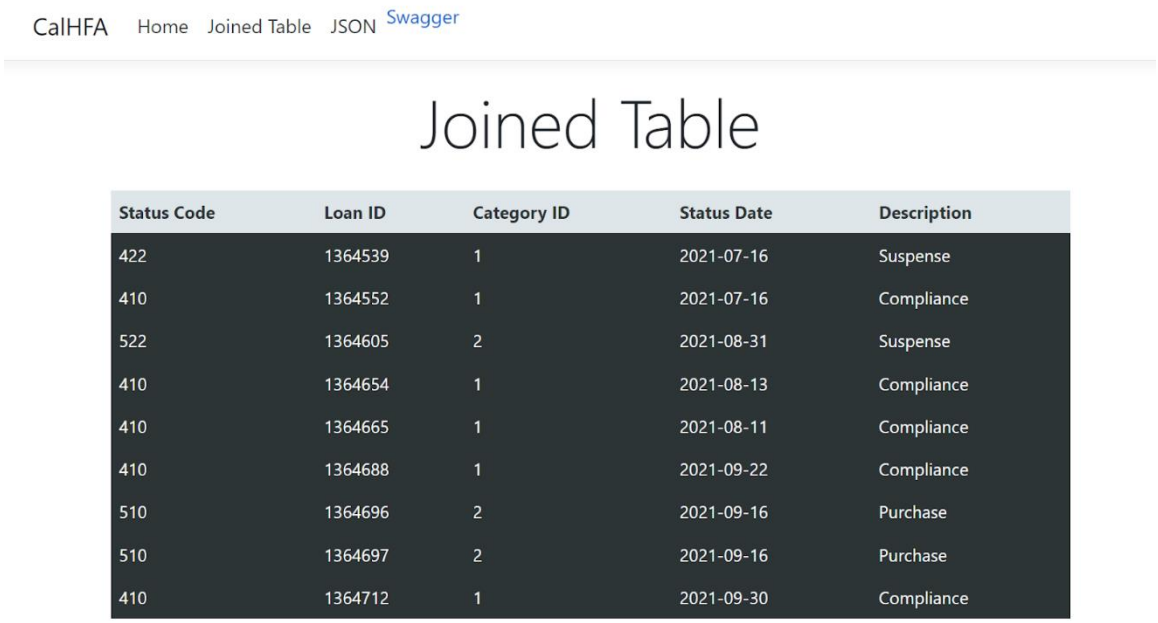



API
Loan ID
Date
Reported Values
Purchase
Suspense
Purchase & Suspense
Compliance
Suspense
Compliance & Suspense

6. User Interface Design

The Interface Design describes internal and external program interfaces. Interface designs are based on the information obtained from the analysis models. Use the Use Case Model and Sequence diagram to capture the interface design. Show menus, submenus, buttons, text boxes, check boxes, down drop lists, links, and tables.

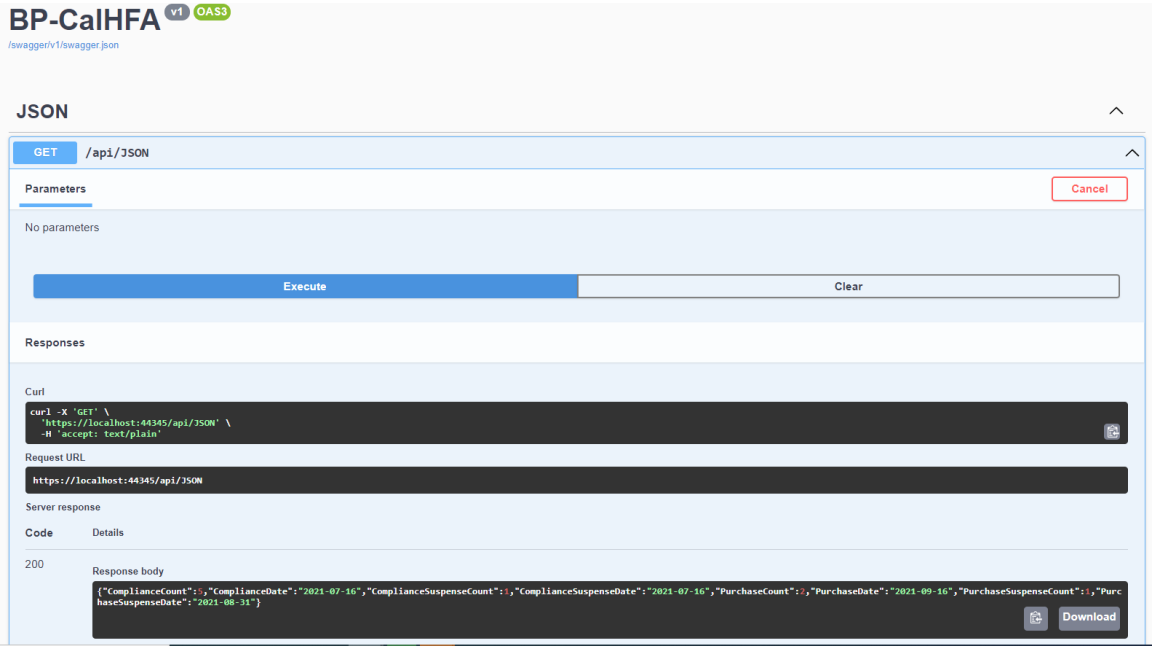
The Home tab in the CalHFA Testing Grounds displays the loan counts and respective dates in a similar fashion to their already existing webpage. Its main purpose is to demonstrate the simplicity in creating a page through our API.



The Joined Table tab displays a view of the resultant table after executing our SQL script. It enables the team to remotely and quickly see additions or removals from the database in real time.

{"ComplianceCount":5,"ComplianceDate":"2021-07-16","ComplianceSuspenseCount":1,"ComplianceSuspenseDate":"2021-07-16","PurchaseCount":2,"PurchaseDate":"2021-09-16","PurchaseSuspenseCount":1,"PurchaseSuspenseDate":"2021-08-31"}

The JSON tab displays the output of the HTTP GET request, in JSON format of the four categories initially requested by the client.



The Swagger tab was included directly into the website for ease of use.

Swagger - ***"Simplify API development for users, teams, and enterprises with the Swagger open source and professional toolset. Find out how Swagger can help you design and document your APIs at scale."***

7. Technology and Tools

a. List the tools that will be used to build the system

- ASP.NET Core Framework
- Entity Framework
- Restful API Architecture style
- Microsoft Visual Studio
- Razor Pages
- Microsoft SQL Server Management Studio
- SQL Express
- MySQL Workbench
- Swagger

8. Assumption and Constraints

Any relevant assumptions and any special design issues, which impact the design or implementation of the software, are noted here.
Assumptions:

- The exact role of the marketing team after the implementation of the API.
- We can assume that all hardware (e.g., server) will be operable.
- We can assume that all software packages will be operable.
- CalHFA team will be able to adjust the API if they need to reflect more loan types or a different time scheme.

Constraints:

- Possible time constraints.