1. Introduction

1.1 Purpose

This document provides details for how our application will be build and aids the development process. It offers a comprehensive architectural overview of the application, using a number of different architectural views to depict different aspects of the application.

1.2 Scope

This document describes the implementation details of our team's (MDZZ) web application. The application consists of 8 micro-services: 2 orchestrators, a *CustomerInfo*, a *CustomerAccount*, a *Catalog*, a *Payment* microservice and two external microservices, *Stripe* and *Amazon Simple Email Service (SES)*. The index page enables users to sign up or login using their email addresses. Registered users could view the items in store, make purchases and check out using their credit cards via Stripe. Users could also check their profiles and accounts (past orders and balance).

1.3 Definitions, Acronyms, Abbreviations

Orchestrator - is understood to be the entity which manages complex cross-domain processes and handles exceptions. Our application implements two orchestrator: one for user registration management and the other for customer account management.

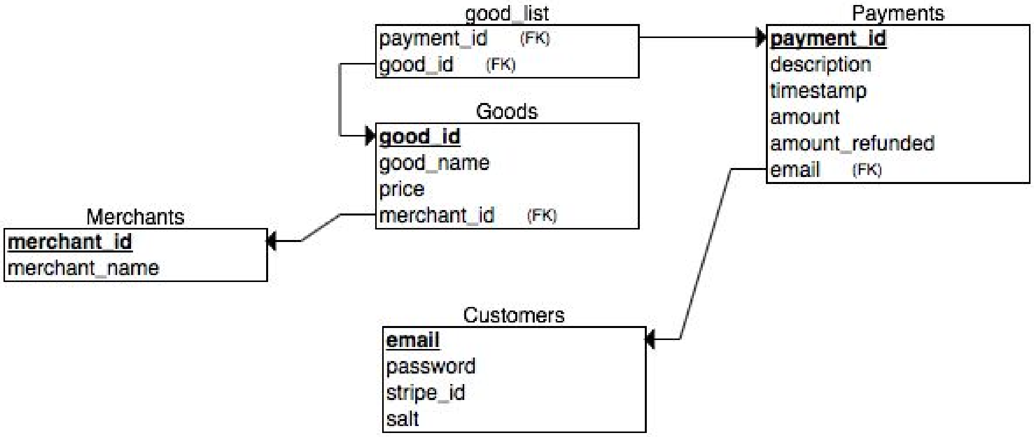
1.4 Overview

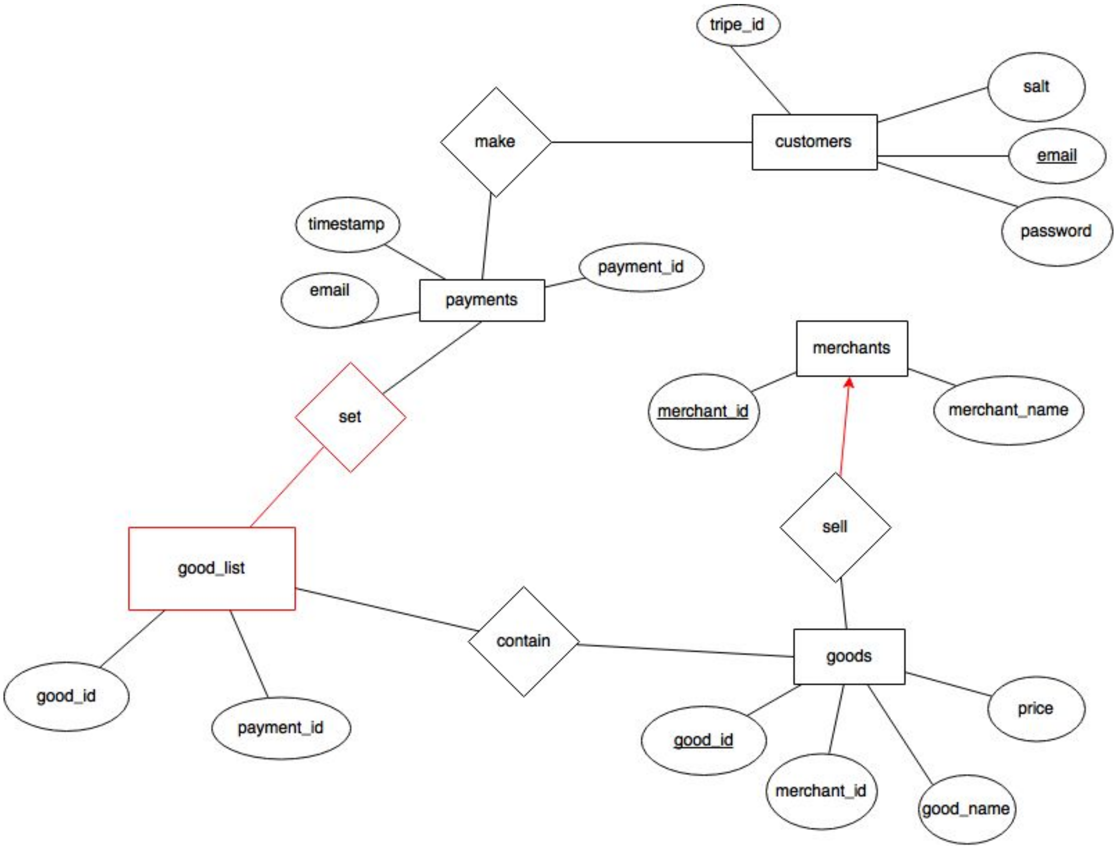
Within the document are narrative and graphical documentations of the software design for the project. This document includes use case models, component diagrams, sequence diagrams, and other supporting information. The document is divided into 6 sections: Introduction, User Stories, Data Model, Component diagram, Interaction Diagrams, Potential flaws and Future Work.

2. User Story

As a / an	I want to	So that
General user	Sign-up / Sign-in via email address	I can ● Get a confirmation email after I sign up ● View available products after I sign-in ● View my user information ● View my account information ● Not register again with an existing email address
General user	Reset password/username if I forget	● Receive email notifications to reset password (optional)
General user	Logout my account of the website (optional)	
General user	Cancel my account (optional)	● Receive an email notifying my cancellation of my account ● Register a new account using the same email address
General user	View a list of products	● View product prices and add to my shopping cart ● Sort the items based on either price or popularity or else (optional)
General user	Submit the order	● View my order and confirm
General user	Make payment	● Pay with my credit card safely ● See an update of my Ordering History (optional) ● Receive an email notification about the payment I made (optional)
General user	View Account	● View the transactions/orders I made in the past. (optional) ● View my account balance
General user	Edit my profile	● Change my user name to a non-used one (optional)
General user	Set my account (optional)	● Change my password ● Change my email address to a non-used one

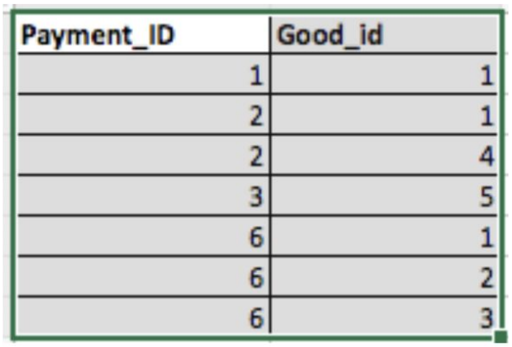
3. Data Model





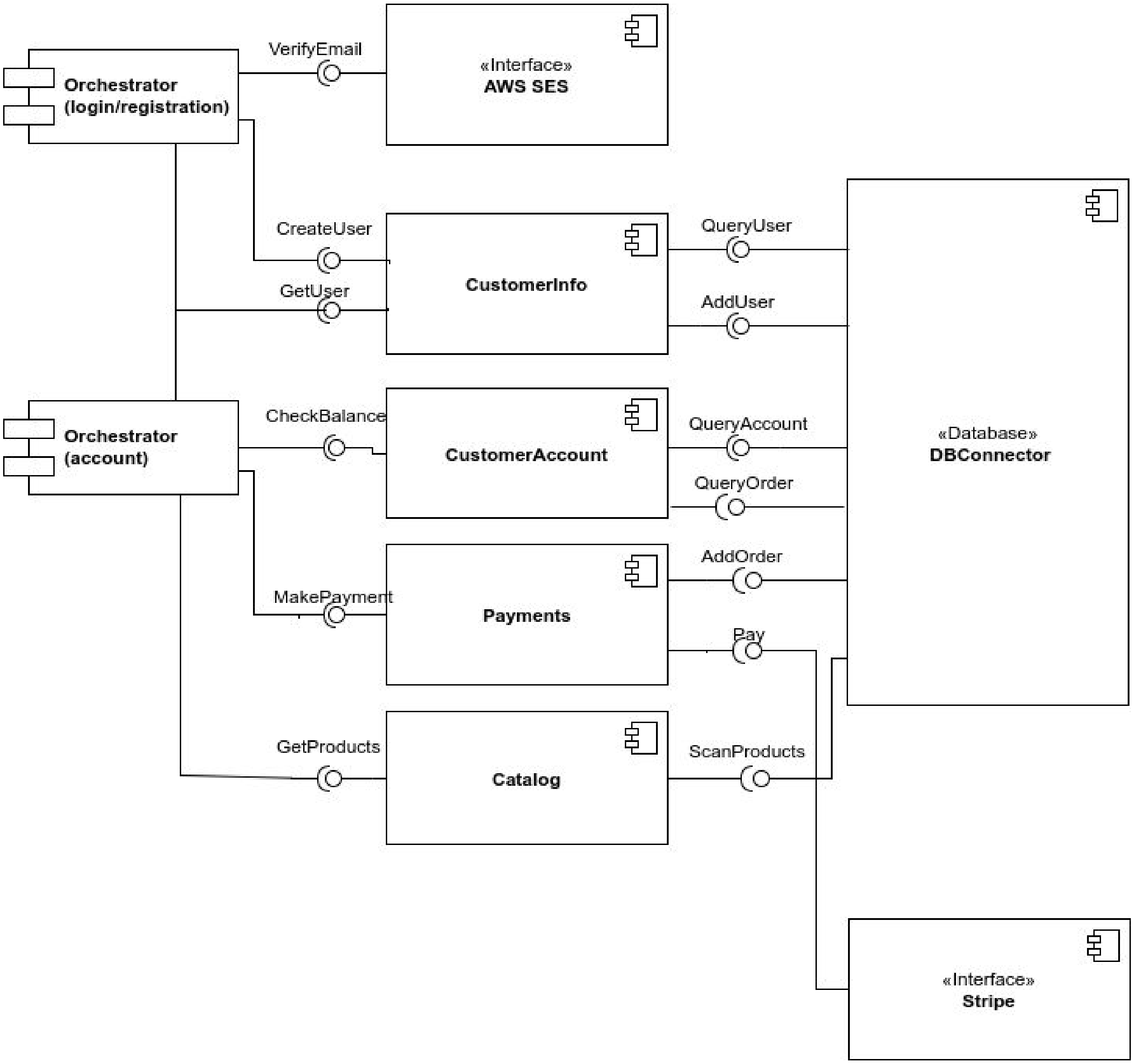
There are four tables in the backend database: *Payments, Good_list, Goods, Customer and Merchant*.

- *Customers* contains core information of the customers and takes email as primary key.
- *Goods* are sold by certain merchants as we set merchant_id as a foreign key.
- *Payments* records both the customer and the goods he/she bought.
- *Good_list* records the list of goods in each payment transaction. (See example below)



Payment_ID	Good_id
1	1
2	1
2	4
3	5
6	1
6	2
6	3

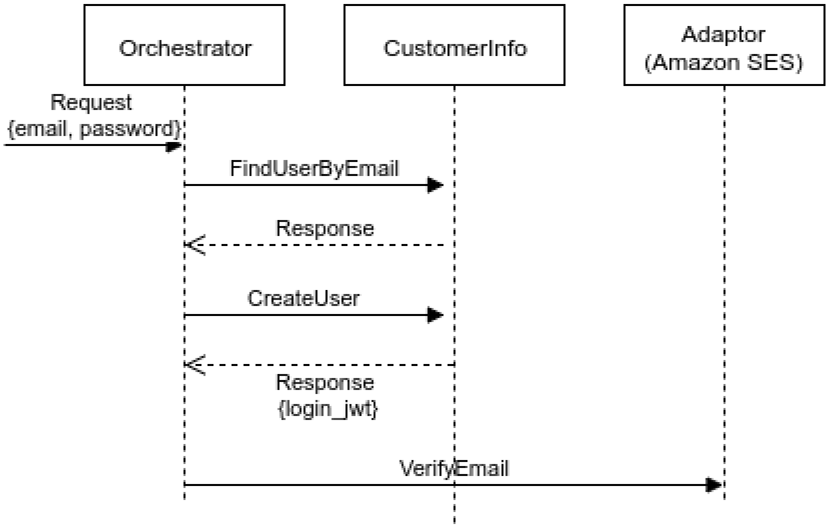
4. Component Diagram

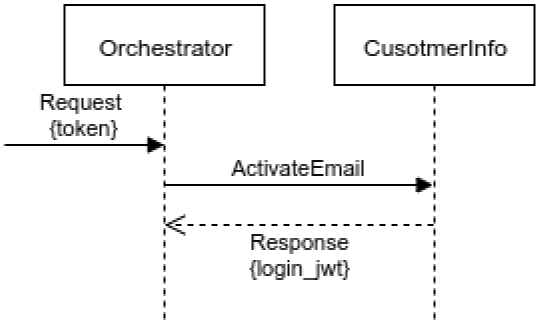


This is the component diagram for our system. We have two orchestrators, one in charge of login and registration event and the other is in charge of general user activity. Behind login orchestrator, we use AWS SES for email verification and CustomerInfo for account management. Behind account orchestrator, we have Account service for purchase history, Catalog service for showing merchandise and Payments service for transaction.

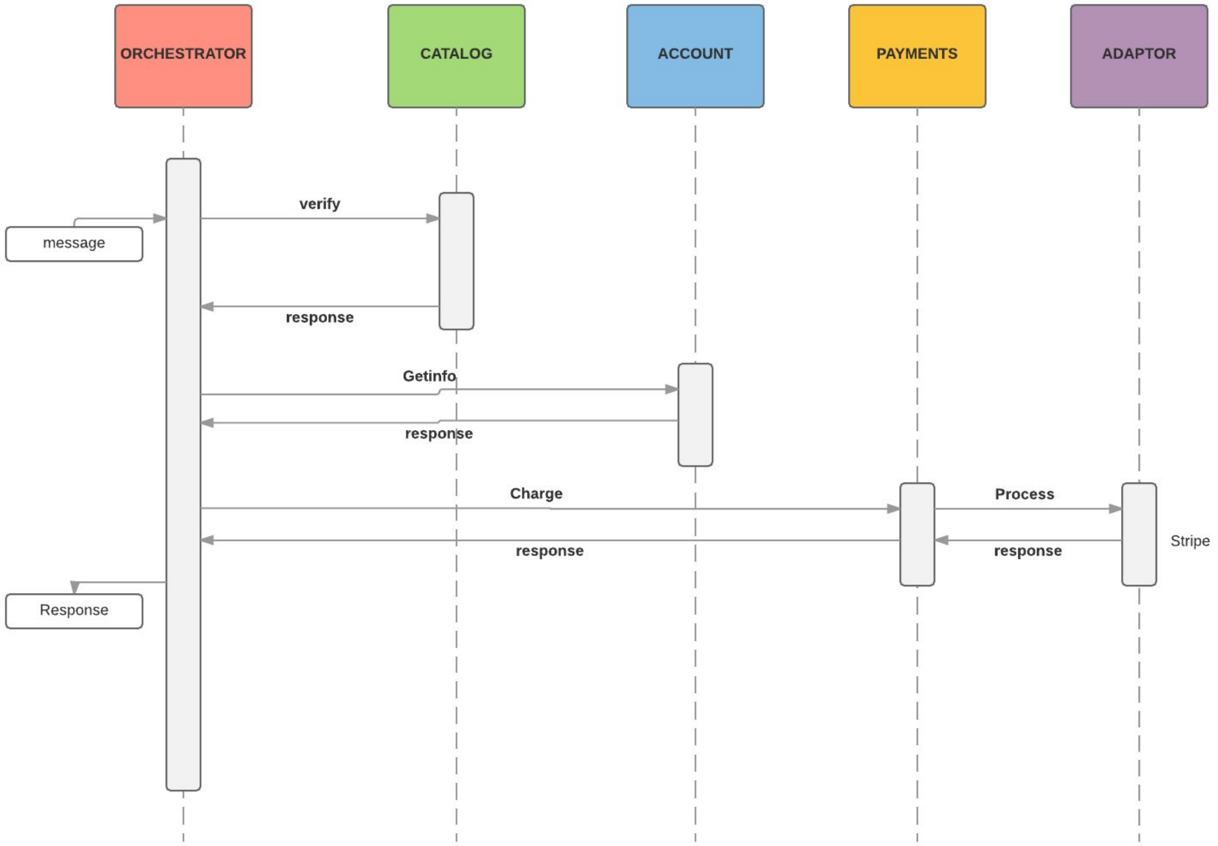
5. Interaction Diagrams

Use Case: New User Registration





Use Case: Make Payment



6. Future Work

In the future, we plan to implement some of the optional features as we mention in the User Case section, such as Logout, Reset Password, and Send notification after purchase.