# Use Cases – Budgeting Program

## Use Case 1 – Log into program

### Description:

A user wants to authenticate with the server to access their account in the program

### Actors:

Customers, Admins, Authentication Service

### Pre-Conditions:

1. The user must be connected to the internet
2. The user must have an account
3. The authentication service on the server must be running

### Post-Conditions:

1. The user is logged into the system and can access their account
   1. If it’s the admin, he will have access to other user accounts as well

### Basic Flow:

1. User (admin or customer) submits their credentials
2. The server creates request and response objects
3. The authentication service tries to match the credentials to what’s stored in the system
4. If it matches the user is granted access to the system
5. If the credentials don’t match, the page displays an error and the user is prompted again – access is not granted

## Use Case 2 – Enter Budget

### Description:

A user wants to enter/edit their budget information

### Actors:

Admins, Customers, Database Service

### Pre-Conditions:

1. User must be authenticated (logged in)
2. The database service must be running

### Post-Conditions:

1. The data that the user inputs is stored in the server database for future use

### Basic Flow:

1. The user enters information into each field on the page
2. When the data is submitted, the database service checks the data to make sure that it’s correct
3. If it’s correct, the data is saved in the user’s fields in the database
4. If it’s incorrect, the page displays errors on the fields with invalid input and the user is prompted again to submit the form