7. Create a Generic pass object

> If your issuer account is in demo mode, you will only be able to issue passes for your test users. If you run the app using a different email address, you will not be able to add the pass to your wallet.

Now that you have verified that the Google Wallet API is available, you can create a pass and prompt your user to add it to their wallet. There are two flows for creating pass objects for users.

Create the pass object on the backend server

In this approach, the pass object is created on a backend server and returned to the client app as a signed JWT. This is best suited for cases where user adoption is high, as it ensures the object exists before the user tries to add it to their wallet.

Create the pass object when the user adds it to their wallet

In this approach, the pass object is defined and encoded into a signed JWT on the backend server. An **Add to Google Wallet** button is then rendered in the client app that references the JWT. When the user selects the button, the JWT is used to create the pass object. This is best suited for cases where user adoption is variable or unknown, as it prevents pass objects from being created and not used. **This approach will be used in the codelab.**

1. Open the `backend/generic_pass.js` file
2. Replace the value of `issuerId` with your Issuer ID from the Google Pay & Wallet Console

```
: Define Issuer ID
uerId = 'ISSUER_ID';
```