TEAM
BEANS

System Operation Descriptions and Activity Diagrams

Wenzheng Kang
Michael Plante
Sabbir Ahmed
Trevor Swann
Ian Wittler

Table of Contents

1. System Operations List

a) Reserve an Audiobook
   i. Authenticate the Customer
   ii. Create Reservation from the requested Audiobook
   iii. Update Reservation status to NEW
   iv. Push new Reservation to queue
   v. Locate Kiosk with the Audiobook available in inventory

b) Request a Title
   i. Authenticate the Customer
   ii. Create Request from the requested Title
   iii. Update Request status to NEW
   iv. Push new Request to queue

c) Check out an Audiobook
   i. Authenticate the Customer
   ii. Offer Customer option to check out Audiobook
   iii. Update Reservation status to FULFILLED
   iv. Pop from Kiosk inventory stack

d) Return an Audiobook
   i. Authenticate the Customer
   ii. Offer Customer option to return Audiobook
   iii. Check for any late fees incurred
   iv. Update Reservation status to RETURNED
   v. Pop from Reservation queue
   vi. Push to Kiosk inventory stack
   vii. Update Request status to FULFILLED
   viii. Pop from Request queue
   ix. Check for open requests for returned audiobook

2. System Operations with State Changes

a) Reservation / Audiobook
   i. Create Reservation from the requested Audiobook
   ii. Push new Reservation to queue
   iii. Pop from Reservation queue
   iv. Update Reservation status to FULFILLED
   v. Update Reservation status to RETURNED

b) Request / RequestQueue / Title
   i. Create Request from the requested Title
   ii. Push new Request to queue
   iii. Pop from Request queue
   iv. Update Request status to FULFILLED

c) Customer
   i. Authenticate the Customer

d) GBALS Kiosk / Audiobook
   i. Push Audiobook to Kiosk inventory stack
   ii. Pop from Kiosk inventory stack

3. System Operation Descriptions

a) Create Reservation from the Requested Audiobook

   i. **Name:** Create Reservation from the Requested Audiobook
   ii. **Description:** Create a new Reservation object
   iii. **Actors:** Customer (p), Reservation
   iv. **Trigger:** Customer reserves an Audiobook
   v. **Inputs:** Title, Customer
   vi. **Classes of objects involved:** Customer, Audiobook, Reservation
   vii. **Precondition:** Audiobook of Title exists
   viii. **Postcondition:** Reservation instance is created with information from Audiobook
   ix. **Outputs:** Return new Reservation instance

b) Push new Reservation to queue

   i. **Name:** Push new Reservation to queue
   ii. **Description:** Push new Reservation object to ReservationQueue
   iii. **Actors:** Customer (p), Reservation
   iv. **Trigger:** New Reservation object is created
   v. **Inputs:** Reservation object
   vi. **Classes of objects involved:** Customer, Audiobook, Reservation
   vii. **Precondition:** Reservation exists
   viii. **Postcondition:** Status of Reservation object is updated to NEW and pushed to ReservationQueue
   ix. **Outputs:** OK on success

c) Pop from Reservation queue

   i. **Name:** Pop from Reservation queue
   ii. **Description:** Pop Reservation object from the Reservation Queue to be fulfilled or to be canceled by the Customer.
   iii. **Actors:** Kiosk (p), Customer, Reservation
   iv. **Trigger:** Customer is ready to cancel or checkout audiobook
   v. **Inputs:** the Reservation to pop
   vi. **Classes of objects involved:** Customer, Title, Reservation, Kiosk
   vii. **Precondition:** Reservation exists within the queue
   viii. **Postcondition:** Reservation no longer is present within the queue
   ix. **Outputs:** A Reservation to either cancel or use to confirm audiobook checkout

d) Update Reservation status to FULFILLED

   i. **Name:** Update Reservation status to FULFILLED
   ii. **Description:** Update status of Reservation at the front of the queue to FULFILLED
   iii. **Actors:** Customer (p), Audiobook (s)
   iv. **Trigger:** Customer checks out Audiobook
   v. **Inputs:** Reservation object, Time
   vi. **Classes of objects involved:** Customer, Audiobook, Reservation
   vii. **Precondition:** Reservation exists, Customer is authenticated
   viii. **Postcondition:** Status of Reservation object is updated to FULFILLED
   ix. **Outputs:** OK on success

e) Update Reservation status to RETURNED

   i. **Name:** Update Reservation status to RETURNED
   ii. **Description:** Update status of Reservation at the front of the queue to RETURNED

iii. **Actors:** Customer (p), Audiobook (s)
iv. **Trigger:** Customer returns Audiobook
v. **Inputs:** Reservation object, Time
vi. **Classes of objects involved:** Customer, Audiobook, Reservation
vii. **Precondition:** Reservation exists, Customer is authenticated
viii. **Postcondition:** Status of Reservation object is updated to RETURNED, Kiosk inventory is updated
ix. **Outputs:** OK on success

f) Create Request for the Requested Title
i. **Name:** Create Request for the Requested Title
ii. **Description:** Create new Request instance
iii. **Actors:** Customer (p)
iv. **Trigger:** Customer requests a title
v. **Inputs:** Title, Customer
vi. **Classes of objects involved:** Customer, Title, Request
vii. **Precondition:** Audiobook of title exists
viii. **Postcondition:** New request instance has been created with information of the title
ix. **Outputs:** Return new request instance.

g) Push new request to queue
i. **Name:** Push New Request to Queue
ii. **Description:** Push new Request object to RequestQueue
iii. **Actors:** Customer (p)
iv. **Trigger:** New Request object is created
v. **Inputs:** Request object
vi. **Classes of objects involved:** Customer, Title, Request
vii. **Precondition:** Request exists
viii. **Postcondition:** Status of Request object is updated to NEW and pushed to the RequestQueue
ix. **Outputs:** OK on success

h) Pop from request queue
i. **Name:** Pop from Request Queue
ii. **Description:** Pop Request object from the Request Queue to be fulfilled or to be cancelled by the customer
iii. **Actors:** Kiosk (p), Customer
iv. **Trigger:** Title becomes available
v. **Inputs:** Available title
vi. **Classes of objects involved:** Customer, Title, Request, Kiosk
vii. **Precondition:** Request exists within the queue
viii. **Postcondition:** Request is no longer present within the queue
ix. **Outputs:** Request object

i) Update Request status to FULFILLED
i. **Name:** Update Request Status to FULFILLED
ii. **Description:** Update status of Request at the front of the queue to FULFILLED
iii. **Actors:** Customer (p), Audiobook (s)
iv. **Trigger:** Title becomes available, and request is popped from queue
v. **Inputs:** Request object
vi. **Classes of objects involved:** Customer, Audiobook, Request
vii. **Precondition:** Request exists
viii. **Postcondition:** Status of Request object is updated to FULFILLED
ix. **Outputs:** OK on success

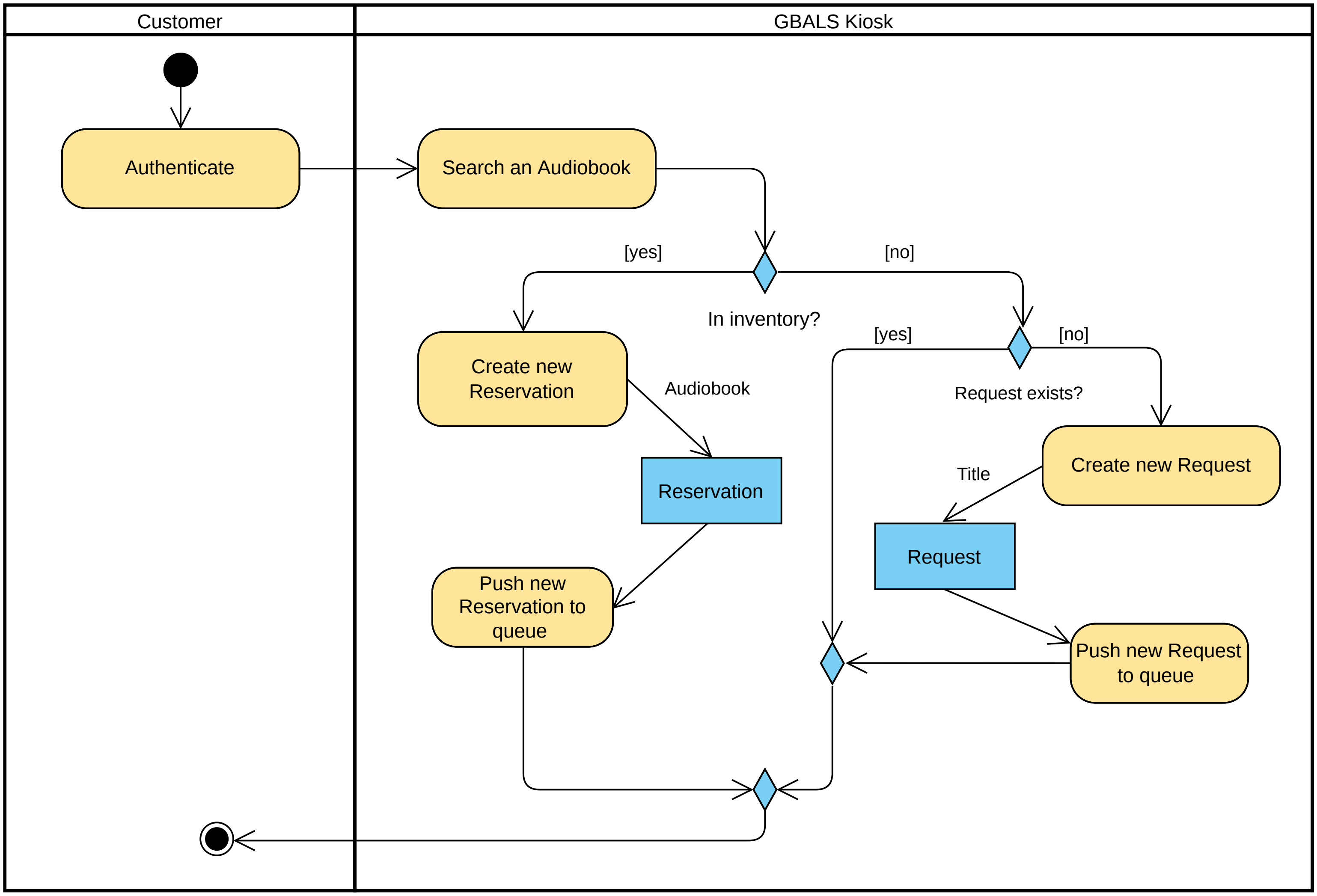
j) Authenticate the Customer

    i. **Name:** Authenticate a customer.
    ii. **Description:** By reaching out to external, already existing, library software a Customer can be authenticated via input information.
    iii. **Actors:** Customer (p)
    iv. **Trigger:** Customer attempts to use Kiosk
    v. **Inputs:** authentication information
    vi. **Classes of objects involved:** Customer and Kiosk
    vii. **Precondition:** N/A
    viii. **Postcondition:** Status of Customer authentication validated
    ix. **Outputs:** Acceptance or denial of Customer into Kiosk

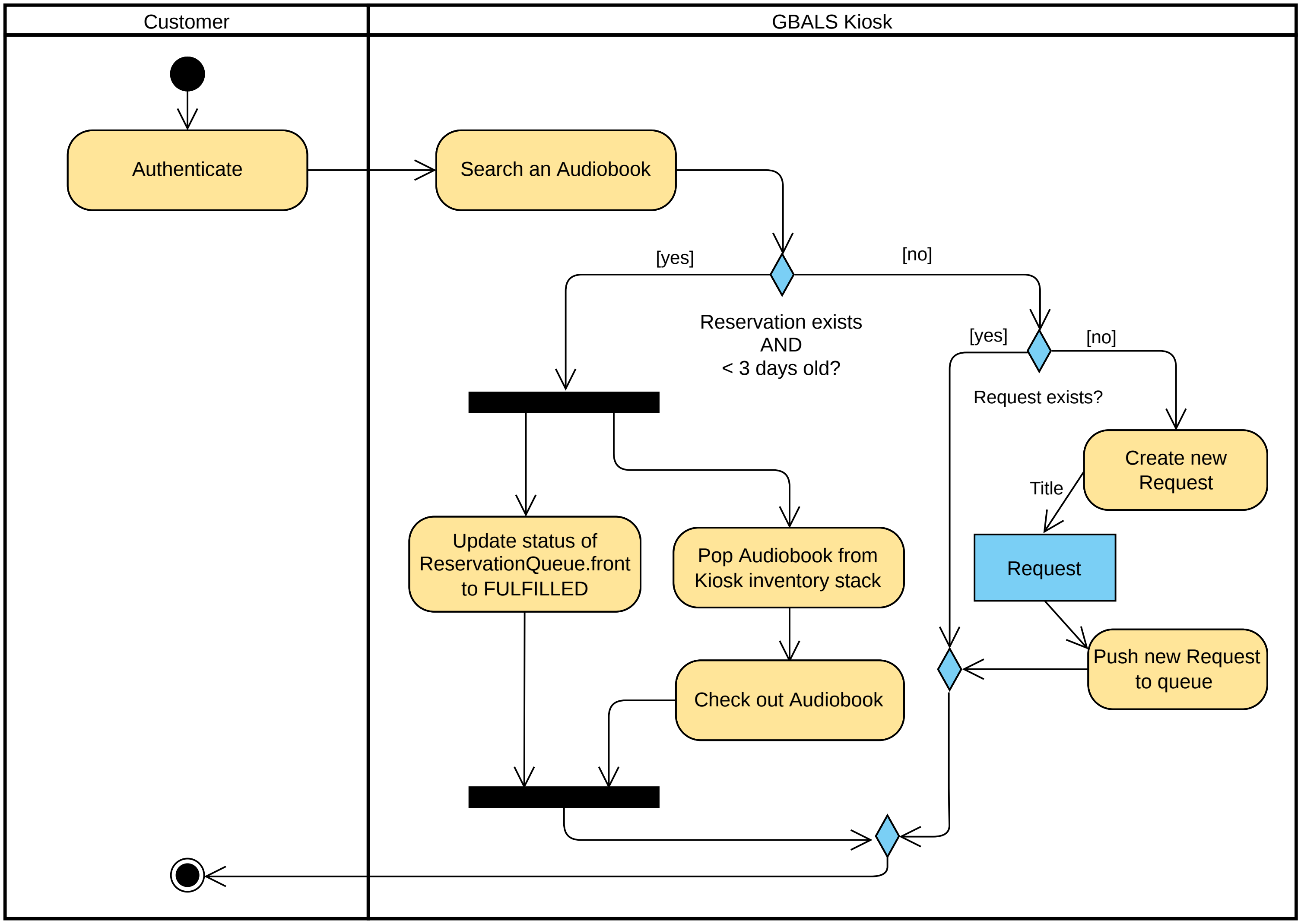
k) Push Audiobook to Kiosk inventory stack
    i. **Name:** Push Audiobook to Kiosk inventory stack
    ii. **Description:** Add an Audiobook object to the Kiosk inventory
    iii. **Actors:** Kiosk (p)
    iv. **Trigger:** Customer returns Audiobook
    v. **Inputs:** Reservation
    vi. **Classes of objects involved:** Customer, Reservation, Kiosk
    vii. **Precondition:** Customer is authenticated
    viii. **Postcondition:** Audiobook is added to the Kiosk inventory
    ix. **Outputs:** OK on success

l) Pop from Kiosk inventory stack
    i. **Name:** Pop from Kiosk inventory stack
    ii. **Description:** Remove last-added Audiobook from Kiosk inventory
    iii. **Actors:** Kiosk (p)
    iv. **Trigger:** Customer checked out Audiobook
    v. **Inputs:** Reservation information
    vi. **Classes of objects involved:** Customer, Reservation, Kiosk
    vii. **Precondition:** Audiobook exists in Kiosk inventory stack, Customer is authenticated
    viii. **Postcondition:** Audiobook is removed from the inventory, Reservation status is updated
    ix. **Outputs:** Audiobook

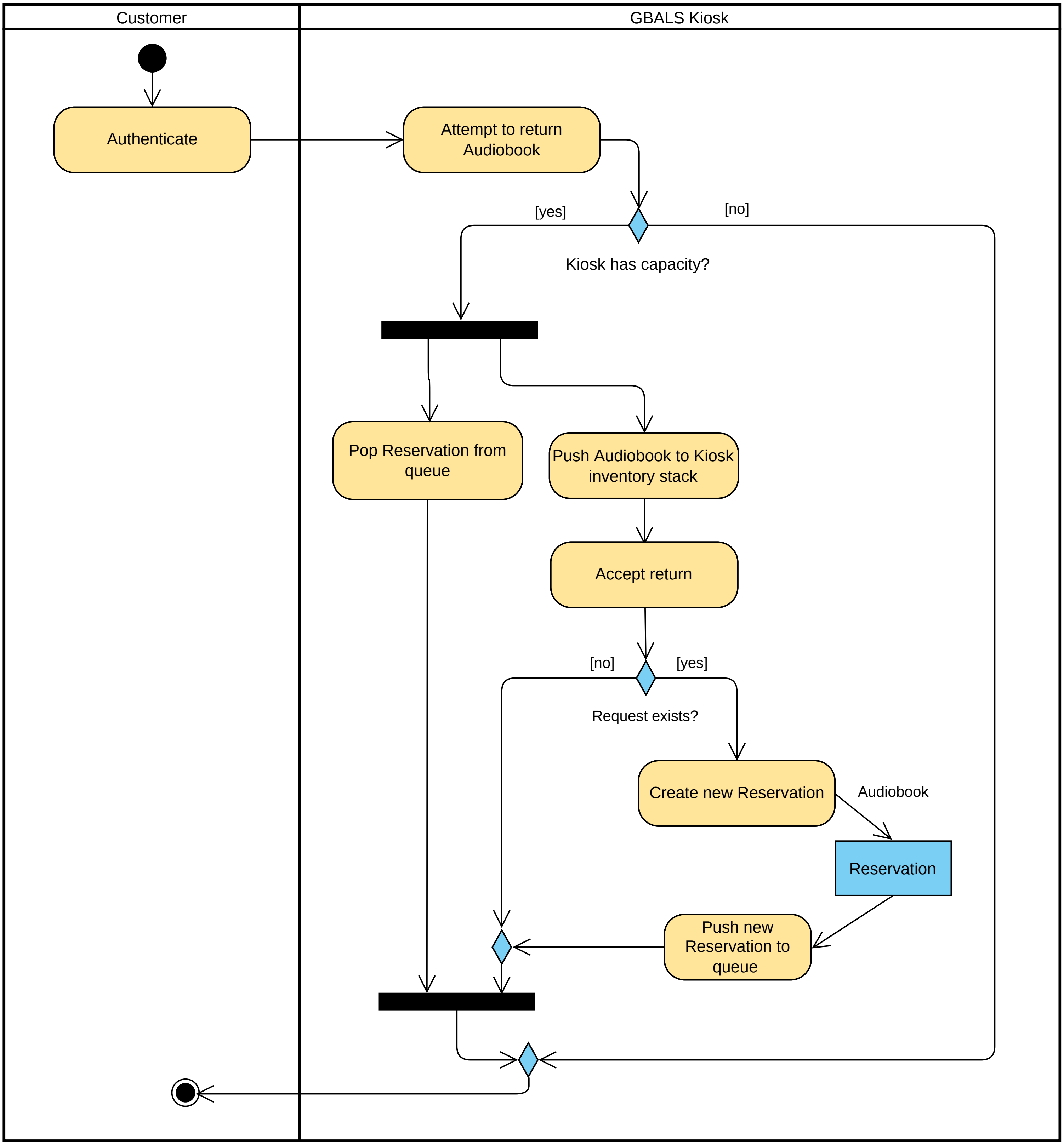
Activity Diagram 1: Request or Reserve an Audiobook



Activity Diagram 2: Check out an Audiobook



Activity Diagram 3: Return an audiobook


Customer
GBALS Kiosk
Authenticate
Attempt to return Audiobook
[yes]
[no]
Kiosk has capacity?
Pop Reservation from queue
Push Audiobook to Kiosk inventory stack
Accept return
[no]
[yes]
Request exists?
Create new Reservation
Audiobook
Reservation
Push new Reservation to queue