**Dan’s Bagel Shop**

**Web App**

**Use Case Diagrams**

*Name:* Add Employee (Chef/Cashier)

*Participating Actors:* Manager, Employee, Database

*Entry Conditions:*

* Manager has met with and decided to hire the new employee
* Proper paperwork has been filled out

*Exit Conditions:*

* Employee has successfully created an account
* Database updates

*Event Flow:*

1. Employee is hired
2. Manager sends link to employee to the employee account creation page
3. Employee creates account
4. Database updates

**Manager**

**Employee**

**Add New Employee**

**Create Employee Account**

**Update database with new account information**

**Database**

**Follow link from Manager**

**Send link to new employee**

<<includes>>

<<includes>>

**Invalid information**

<<extends>>

*Name:* Order Inventory

*Participating Actors:* Manager, Database

*Entry Conditions:*

* Manager logs in and navigates to inventory page

*Exit Conditions:*

* Manager orders inventory
* Database updates

*Event Flow:*

1. Manager navigates to inventory page
2. Manager selects items and quantities to order
3. Manager clicks the “Order” button
4. Database updates

**Manager**

**Navigate to Inventory Page**

**Select items and quantities to order**

**Order Inventory**

<<includes>>

**Database**

**Update inventory in Database**

*Name:* Edit Menu

*Participating Actors:* Manager, Database

*Entry Conditions:*

* Manager logs in and navigates to menu page

*Exit Conditions:*

* Manager clicks “save”
* Database updates

*Event Flow:*

1. Manager navigates to menu page
2. Manager selects ‘edit’
3. Manager makes desired changes
4. Manager clicks ‘save’
5. Database updates

**Manager**

**Navigate to Menu Page**

**Save changes**

**Edit Menu**

<<includes>>

**Database**

**Update Menu in Database**

*Name:* Create Account

*Participating Actors:* Customer, Database

*Entry Conditions:*

* Customer navigates to Dan’s Bagel Shop web page

*Exit Conditions:*

* Customer clicks ‘save’
* Database updates

*Event Flow:*

1. Customer navigates to Dan’s Bagel Shop web page
2. Customer selects ‘create account’
3. Customer follows instructions (inputting personal information)
4. Customer clicks ‘save’
5. Database updates

**Customer**

**Navigate to main web page**

**Save**

**Create Account**

<<includes>>

**Database**

**Update account information in Database**

**Invalid information**

<<extends>>

*Name:* Order Bagel

*Participating Actors:* Customer, Database

*Entry Conditions:*

* Customer selects ‘order bagel’

*Exit Conditions:*

* Customer selects ‘place order’
* Database updates

*Event Flow:*

1. Customer selects ‘order bagel’
2. Customer gives order information
3. Customer clicks ‘place order’
4. Database updates order history

**Customer**

**Select ‘order bagel’**

**Log in to account**

**Select order details**

<<includes>>

**Database**

**Update order history in Database and add bagel to queue**

**Item not available**

<<extends>>

**Incorrect log in information**

<<extends>>

**Select ‘place order’**

*Name:* Add Money to Account

*Participating Actors:* Customer, Database

*Entry Conditions:*

* Customer selects ‘Add money’ in the account information page

*Exit Conditions:*

* Customer selects ‘confirm’
* Database updates

*Event Flow:*

1. Customer navigates to account information page
2. Customer selects ‘add money’
3. Customer selects an amount to add
4. Customer clicks confirm
5. Database updates account balance

**Customer**

**Navigates to Account Information page**

**Log in to account**

**Select ‘add money’**

<<includes>>

**Database**

**Update account balance in Database**

**Incorrect log in information**

<<extends>>

**Select amount to add**

**Confirm amount**

<<includes>>

*Name:* Bagel Creation

*Participating Actors:* Chef, Database

*Entry Conditions:*

* Chef starts making a bagel in the queue

*Exit Conditions:*

* Chef finishes the bagel
* Database updates

*Event Flow:*

1. Chef selects the first order in the queue
2. Chef makes the bagel
3. Chef changes order progress to ‘finished’
4. Database updates order progress

**Chef**

**Set order progress for first order in queue to ‘in progress’**

**Make Bagel**

**Database**

**Update order status in Database**

**Set order progress to ‘finished’**

*Name:* Bagel Creation

*Participating Actors:* Cashier, Customer, Database

*Entry Conditions:*

* Bagel is finished
* Customer arrives to pick up order

*Exit Conditions:*

* Customer receives order
* Database updates

*Event Flow:*

1. Bagel is finished
2. Customer arrives to pick up order
3. Cashier verifies order details
4. Customer receives order
5. Database updates order progress

**Customer**

**Enter Dan’s Bagel Shop**

**Verify order details**

**Database**

**Update order status in Database**

**Give order to Customer**

**Cashier**

**Set order status to ‘picked up’**