**Inwood Bagels Website**

**Iteration 4 Use Case Specifications**

**Revision History**

| **Date** | **Revision #** | **Created By** | **Revision Notes** |
| --- | --- | --- | --- |
| 04/03/2022 | 1.0 | Gaëlle Gilles | Initial Draft |
| 04/08/2022 | 2.0 | Gaëlle Gilles | Added Use Case Specification 01.06 Update Admin Account |
| 04/10/2022 | 3.0 | Noah, Brinda, Hemaswi, Nandu | Updated alternative flows, exceptions, and alternative flows. Add JP |
|  |  |  |  |
|  |  |  |  |

**UC. 1.07 Update user account details (Brinda)**

| **Actor(s):** | Customer |
| --- | --- |
| **Short Description:** | This use case will allow customers to change information like email address, password, and credit/debit card information |
| **Preconditions:** | User must already have an account |
| **Postconditions:** | Account will reflect the necessary changes |
| **Frequency of Use:** | Low |
| **Normal Flow of Events:** | |
| 1. **Actor** clicks on the login button 2. **The system** redirects them to the login page **[JP 1: ET]** 3. **Actor** logs in to their account with their respective email address and password 4. The **actor** clicks on the account tab in navigation and the **System** redirects them to account page 5. **The actor** clicks on the update account button and updates their account information. 6. The **actor** clicks on the save button **[JP 3: CN]** 7. The **system** saves updated information on their account page and updates the database as well. This use case ends here [**JP 2: DFO]** | |
| **Alternative Flows:** | |
| none | |
| **Exceptions:** | |
| E1. If the actor doesn’t want to save changes, after step 5:   1. The **actor** will click on the cancel button 2. The **system** does not update account 3. Use case ends here | |
| **<<Include>> Relationships:** | None |
| **<< Extend>> Relationships:** | None |
| **Business Rules:** |  |
| **Assumptions:** |  |

**UC.02.02 Edit Menu Item Cost/Description (Hemaswi)**

| **Actor(s):** | Business Owner, Day manager |
| --- | --- |
| **Short Description:** | The person on the admin side should be able to change an items cost or description whenever needed |
| **Preconditions:** | Admin user should have an account and be logged in to account in order to make changes to the menu items |
| **Postconditions:** | Admin user is able to update the menu items on the website. |
| **Frequency of Use:** | Medium |
| **Normal Flow of Events:** | |
| 1. **Actor** logs into account **[JP1: ET]** 2. **Actor** selects an item to be changed 3. **Actor** clicks on the edit menu item button 4. **Actor** changes the price and/or description of the menu item **[JP2: DDV, FV]** 5. **Actor** clicks the save button **[JP3: SI-In, CN]** 6. **System** reflects the changes to the customer side of the menu and the database. The use case ends **[JP4: DF-Out]** | |
| **Alternative Flows:** | |
| None | |
| **Exceptions:** | |
| E1. If the actor does not want to save changes, after step 4   1. The **Actor** clicks the cancel button if the changes are not intended to be made. 2. The **System** does not reflect any changes to the menu or database and the use case ends **[JP4: DF-Out]** | |
| **<<Include>> Relationships:** | None |
| **<< Extend>> Relationships:** | None |
| **Business Rules:** |  |
| **Assumptions:** |  |

**UC. 05.01 Add credit/debit card (Noah)**

| **Actor(s):** | Customer |
| --- | --- |
| **Short Description:** | Customer has the ability to add a credit or debit card |
| **Preconditions:** | Actor must be logged into their account |
| **Postconditions:** | A user views the add card section |
| **Frequency of Use:** | High |
| **Normal Flow of Events:** | |
| 1. **Actor** must log in **[JP 1: SI-Out, ET, DDV]** 2. **System** will validate the account **[JP 2: Authorization, ET]** 3. **Actor** will add their credit/debit card information in the add new payment section 4. **System** will validate the credit/debit card **[JP 3: Authorization, ET, Connectivity]** 5. **System** will save the **Actor**’s information into the database **[JP: 4 DFO, ET]** | |
| **Alternative Flows:** | |
| A1. Invalid Credit/Debit Card Information. Occurs after step 3   1. If the **Actor** enters invalid credit/debit card information a pop up will say ‘Invalid information’ | |
| **Exceptions:** | |
| E1. Cancel adding credit/debit card information. Occurs after step 3   1. **Actor** clicks on the cancel button. 2. **System** does not validate credit or debit card and the **actor**’s information is not saved in the database. | |
| **<<Include>> Relationships:** |  |
| **<< Extend>> Relationships:** |  |
| **Business Rules:** |  |
| **Assumptions:** |  |

**UC. 01.06 Update Admin Account Details (Nandu)**

| **Actor(s):** | Business Owner, Day Manager, Accountant, Purchasing Officer |
| --- | --- |
| **Short Description:** | Actor will have the ability to change their account information such as email address and password |
| **Preconditions:** | Business owner should have an admin account and be logged in to update their account |
| **Postconditions:** | Account should reflect any changes that the actor has made |
| **Frequency of Use:** | Low |
| **Normal Flow of Events:** | |
| 1. **Actor** clicks on the login button 2. The **system** redirects them to the login page **[JP 1: ET]** 3. **Actor** logs in to their account with their respective email address and password 4. The **actor** clicks on account tab in navigation 5. The **system** redirects them to the account page 6. The **actor** clicks on update account button and updates the information that he/she wants to update. 7. **System** saves updated information to not only their account page but database as well [**JP 2: DFO]** 8. **Actor** clicks on save button **[JP 3: CN]** | |
| **Alternative Flows:** | |
| None | |
| **Exceptions:** | |
| **Exceptions:**  E1. The **actor** doesn’t want to save changes, after step 6   1. The **Actor** clicks on the cancel button 2. The **system** does not save any change information to the actor’s account and database | |
| **<<Include>> Relationships:** |  |
| **<< Extend>> Relationships:** |  |
| **Business Rules:** |  |
| **Assumptions:** |  |