|  |  |
| --- | --- |
| *Use Case Name* | UC\_03\_UpdateAccount |
| *Participating Actor* | Student |
| *Entry Condition* | Student should be logged into their account to be able to update their account. |
| *Flow of Events* | 1. Update first name, last name, email, and password, major.  2. Check if email already exists in database. If it does, inform student that email is already in use.  3. Update remaining fields into database.  4. Redirect student back to their timeline.  5. Alert student that their account has been updated. |
| *Exit Condition* | Once all parameters have been verified, then the system will redirect the student back to their timeline. All information will be updated in the database for the student. |

|  |  |
| --- | --- |
| *Test-Case Identifier* | UC\_03\_UpdateAccount |
| *Test Location* | /update |
| *Feature to be Tested* | Update existing attributes on user account |
| *Feature Pass/Fail Criteria* | The test passes if the user submits the updated information into the form and the account reflects the new changes. |
| *Means of Control* | Student() enters information into the fields, Booki() creates new database objects with information, Booki() directs user to their newly updated site. |
| *Data* | The data will be entered by the user the inserted into the database |
| *Test Procedure* | 1. User has an account 2. User has existing information in the account 3. User is prompted to enter new information 4. User’s input is checked for validation 5. The user is directed to their account with the new information |
| *Special Requirements* | User must have an existing account |
| *Rational* | 1. user has incorrect account information 2. user need to update their account informtation |