**Business Goals**

1. Register User
2. Set the account status to trying
3. Store
   1. Store and encrypt some fields
4. Separate process to activate account
5. Signup
6. After Authentication
7. User Actions
8. Success Account Validation
9. Trigger Scrapping

**User Scenarios**

**Introduction.**

1. Customer view account online.

8 Provide single access for customer to view accounts.

**Website Scraping**

5 Uses User credentials

**Error Conditions**

0 On error generate error XML

Invalid credentials -> 5 Change status from active to inactive

-> 5 present invalid logon

-> 5 provide updated credentials or logon again

-> 2 if user has updated details change status to trying.

**User not signed up**

1. Register User

4 Separate process to activate e-billing

5 If customer not signed up- prompt sign-up

**Action Required by Billing Company’s Website**

The customer should action the notifications below

6 Update details -> customer email.

6 Customer to accept terms and conditions

**Credential Management**

Customer to provide.

3 Strored, handled, managed Securely

**Credential Types**

* Identification fields
* Authentication fields
* 3.1 Enter twice during registration
* 3.1 Store encrypted
* 5 Decrypt during authentication and discard the clear text
* 3.1 Encrypt date of birth, credit card – decrypt when displaying
* 3.1 APS Cater for 5 plain text and 5 encrypted. Allow for maximum length of 100

**Incorrect Account**

7 After logon prompt for account.

8 Validate account before scrapping if not the same abort.

**Scope and Assupmtions**

**3** encryption for Customer credentials

**User Stories and Acceptance Criteria**

1.

|  |  |
| --- | --- |
| As a | User |
| I want to | Register |
| So that | I can logon |

|  |  |
| --- | --- |
| Given | User clicks link |
| When | Wants to view accounts |
| Then | Route to Register if user not in database |

2.

|  |  |
| --- | --- |
| As a | System |
| I want to | Set the status to trying |
| So that | User can change status to active |

|  |  |
| --- | --- |
| Given | User is registering for the first time |
| When | Wants to be a subscriber |
| Then | Set the status to trying |

3.

|  |  |
| --- | --- |
| As a | System |
| I want to | Store and encrypt user information |
| So that | User can exist in Data Base |

|  |  |
| --- | --- |
| Given | User information is being stored |
| When | User pass input validation |
| Then | Encrypt some information before storing |

4.

|  |  |
| --- | --- |
| As a | System |
| I want to | Use a separate process to activate account |
| So that | User can activate account |

|  |  |
| --- | --- |
| Given | User tries to activate account |
| When | Captured information is stored successfully |
| Then | Change the status to active |

5.

|  |  |
| --- | --- |
| As a | User |
| I want to | Logon |
| So that | I can view accounts |

|  |  |
| --- | --- |
| Given | User tries to view accounts |
| When | Clicking a link |
| Then | Check if the user is in the database, logged on, or provided correct credentials. Authenticte. |

6.

|  |  |
| --- | --- |
| As a | System |
| I want to | Let the user know about changes |
| So that | User can take action on the notifications |

|  |  |
| --- | --- |
| Given | Successful Authentication |
| When | User trying to enter account number |
| Then | Check for system notifications and user to action |

7.

|  |  |
| --- | --- |
| As a | System |
| I want to | Prompt the user to enter account number |
| So that | After authentication I can validate for more security |

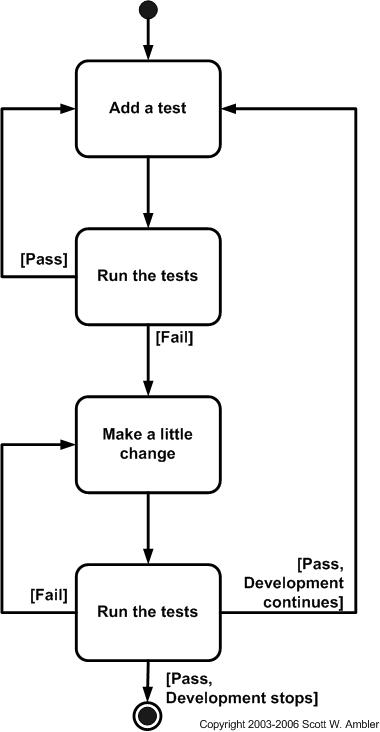
|  |  |
| --- | --- |
| Given | Successful Authentication and accepted system notifications |
| When | Viewing accounts |
| Then | Add account number |

8

|  |  |
| --- | --- |
| As a | User |
| I want to | Provide credentials |
| So that | To see account information |

|  |  |
| --- | --- |
| Given | Successful authentication and account number validation |
| When | User view information |
| Then | Display Account information |

**TDD Steps**



Source: http://www.agiledata.org/essays/tdd.html

**Unit Tests**

1. **Register User Unit Tests**

User not in Database -> Create query Module

User Registered Successfully -> Write a user registration module

User in Database -> Create query Module

User Status is not Active

Check input Length

1. **Set the account status** **Unit Tests**

Is user status is active

Is user status is inactive

Is user account trying

1. **Strored, handled, managed Securely**

Some User fields should be encrypted

Encrypted Fields

Decrypt Fields

1. **Separate process to activate account**

Check if user exist

Check account status

Notify user about activation process

After use took action Activate account

Check account status

1. **Signup** **Unit Tests**

Username and password not null

Check if user already logged on

Check if user in database

Decrypt user credentials

Authenticate customer

Check if authentication valid

1. **After Authentication** **Unit Tests**

Check for system notification

Type of notification

1. **User Actions** **Unit Tests**

Prompt for account number

Validate account number entry

1. **Success Account Validation** **Unit Tests**

Check if account belongs to user

1. **Schedule Scrapper**

**Additional development requirements**

Below is extra required development to fulfil the user scenario.

1. Register User

Database interaction

Validation framework

1. Store and handle information.

Encrypt information

Decrypt information

1. Process to activate account

Email notification or file notification

1. Check for system notifications

Email update

Accept new t & Cs