Sprint One Stories

**Story line 1:**

As a customer, I want the ability to see a help option so that I will know how to operate the program without any conflicts. **(1 story point)**

**Assumptions:** A main window page will be present so that the customer can see the help button

**Tasks:**

* Using QT, the main window’s UI will be modified such that a help button will be displayed after the program’s execution.
* After the user clicks the help button, a new page will open which will assist the customer in dealing with the program
* We will create a function so that after the help button is clicked, a new page opens and the current widget will be set to that specific page.
* In the help page, a “Return to home” menu will also be displayed which will let the user return back to the home page

**Definition of Done:**

* Make sure the code adheres to the coding standard
* Make sure that the finished story meets the objectives
* Document the code to explain the meanings/implementations behind certain areas of code
* Test the code so that it meets the required goals
* Confirm with the product owner to see if it works
* Gain approval from the customer, if the customer does not approve on certain aspects of the story, then refine it/meet the new objectives

**Story line 2:**

As an administrator, I would like to contain a sales pitch that includes key selling points of I Cybersecurity Inc keeping the target market in mind. The sales pitch will incentivize the customer to really purchase our products and select the most suitable one for them. **(1 story point)**

**Assumptions:** A main window page will be present so that the sales pitch is visible and is the first thing that the customer sees.

**Tasks:**

* Using QT, the main window’s UI will be modified such that a text box is displayed which contains the sales pitch
* A text box will be dragged to the center of the main page and a paragraph or two will be written which has all the sales pitch including key selling points of iCybersecurity Inc

**Definition of Done:**

* Make sure the code adheres to the coding standard
* Make sure that the finished story meets the objectives
* Document the code to explain the meanings/implementations behind certain areas of code
* Test the code so that it meets the required goals
* Confirm with the product owner to see if it works
* Gain approval from the customer, if the customer does not approve on certain aspects of the story, then refine it/meet the new objectives

**Story line 3:**

As a customer, I want to have the ability to view a list of environments that will be supported with I Cybersecurity so that I can pick the right one for my computer (**1 story point)**

**Assumptions:** A help option in the main window will contain the instructions on how to operate the program as well as the list of environments that will be supported (Windows, Linux, etc)

**Tasks:**

* Using QT, after the “help” button is clicked and the help page is opened, the instructions will contain the list of environments that are supported
* This story will use the similar task in story #1, where after the user clicks the “help button,” a function will be called so that the page changes to the help options page
* There will be a separate connections function so that the built in connect function will connect the signal and the slot. In this case, the signal is the help button being clicked, and the slot is the page being changed to the help page
* Furthermore, the help button will have the option to let the user return back to the main menu (as in Story #1)

**Definition of Done:**

* Make sure the code adheres to the coding standard
* Make sure that the finished story meets the objectives
* Document the code to explain the meanings/implementations behind certain areas of code
* Test the code so that it meets the required goals
* Confirm with the product owner to see if it works
* Gain approval from the customer, if the customer does not approve on certain aspects of the story, then refine it/meet the new objectives

**Story line 4:**

As a user when I open the program, I want to always have a manual that explains how the program works to make it easy to navigate. **(1 Story Point)**

**Assumptions:** When the user opens the program immediately, he will see what I cybersecurity does and how it works.

**Tasks:**

* QT will be used as our main application.
* The concept of operations will be displayed first thing to the user.

**Definition of Done:**

* Code is tested by the team.
* Code is commented.
* Make sure that the code follows our naming conventions.
* Show the work to the product Owner.
* Add it to the master file.
* Get feedback from Customer.

**Story line 5:**

As a user I want to be able to see different products and packages for to give me more options and make sure to pick the product that will fit my needs. **(2 Story Point)**

**Assumptions:** When the customer decides to buy the product, he will be faced with three options, three prices and three maintenance plans.

**Tasks:**

* Using QT a page will be created specifically for products display.
* A check box will be available by the products that will give the user the ability to pick more than one product.
* Buttons will be programmed to go back or to advance with the purchase.
* Offer a detailed text that explains the difference between the different products
* Add necessary buttons.

**Definition of Done:**

* Code is tested to work as intended.
* Make sure all the buttons are working.
* Peer review it.
* Make sure code is commented.
* Make sure the product is tested by the Product owner.
* Get feedback from the customer.
* Make sure the details are sufficient for user understanding.
* Make sure code adheres to coding standards.

**Story line 6:**

As a user I want to be able to understand the difference between the different products offered by I cybersecurity to be able to make the right choice. **(1 Story Point)**

**Assumptions:** When the customer decides to buy the product, a detailed paragraph will show by every product that explains different logistics and maintenance plans.

**Tasks:**

* Create an add to cart page.
* Add three Text browser and type in the logistics and maintenance plans
* Create all the different connections between different pages.
* Create the buttons to go back and forward.

**Definition of Done:**

* Make sure the buttons are working properly.
* Make sure the page design is easy to understand.
* Get feedback from customer.
* Make sure code adheres to the coding standards.
* Make sure code is commented.
* Peer review it and make sure everybody understands it.
* Make sure the check buttons work as intended.

**Story line 7:**

As a customer, when I’m considering buying a product plan, I would like to be knowledgeable of the guarantee policy. **(2 Story Point)**

**Assumptions:** An information button that is accessible to the customer which lists the variant guarantee policies.

**Tasks:**

* Will be done in the application QT
* Create a push button that will be only be accessed/visible if the customer pushes the buy button
* Create a new page dedicated to the guarantee policy
* Create text boxes to display the guarantee policy
* Implement a push button function so that it will show the guarantee policy page
* Create an option to exit the guarantee policy menu and return to the buy product plans page

**Definition of done:**

* The customer is able to visibly see the guarantee policy button when he/she plans to buy a product plan
* The guarantee policy button displays the guarantee policy and nothing else
* The guarantee policy’s text is visible to the customer, the customer is not allowed to make edits over the policy
* The push button in which to exit the guarantee policy page is visible to the customer
* The push button in which to exit the guarantee policy page returns the customer to the buy product plans page and not to the home page

**Story line 8:**

As a customer, I would like to view the customer reviews on your product plans. It would help me solidify my decision of purchasing the plan and it would be provided an unbiased opinion on your products. **(1 Story Point)**

**Assumptions:** A button which is accessible to the customer in which it provides them the different customer testimonials.

**Tasks:**

* Will be done on the application QT
* Create a push button that is located on the home page that is accessible and visible to the customer
* Create a new page that is dedicated to the customer testimonial page
* Create two text boxes which displays the customer reviews
* Type in the customer reviews for each text box
* Create a back button on the customer testimonial page
* Implement code for the back button so that the customer could access the home page once more
* Implement code for the push function which goes directly to the customer testimonial page

**Definition of done:**

* The customer is able to visibly see the customer testimonial button on the home page. The button should not be visible anywhere else.
* The testimonial button displays the customer testimonials and nothing else.
* Once on the customer testimonial page, the testimonial text is visible to the customer
* The customer can’t have the ability to edit the customer testimonial page.
* The back button located on the customer testimonial page returns the customer only to the home page and nowhere else.

**Story line 9:**

As a customer, in the scenario where I have a question or problem with your product plans, I would like to know the different methods to contact your company. **(1 Story Point)**

**Assumptions:** Creating a push button in which shows the customer the different ways he/she could contact our company.

**Tasks:**

* Will be done on the application QT.
* Create a push button that is located on the home page which is visible and accessible to the customer.
* Create a new page dedicated to the contact page.
* Create text boxes for the information regarding emails, fax, phone numbers etc..
* Type in the appropriate information for each of the text boxes (email: [iCyberSecurity@gmail.com](mailto:iCyberSecurity@gmail.com), etc..).
* Create a back button on the contact page.
* Implement code for the push button so that it goes directly and only to the contact page.
* Implement code fo the back button so that it goes back to the home page.

**Definition of done:**

* The customer is able to visibly see the contact page which is only accessed on the home page.
* The contact button displays the contact page and nothing else.
* Once the customer has access to the contact page via contact button, the customer should be able to see the text.
* The customer can’t edit the contact page information.
* The back button located on the contact page is visible and usable. For usability, it is expected that it returns and only returns to the home page.