|  |  |
| --- | --- |
| **Applicant’s Name:** | Piotr Krajewski |
| **Date Exam Taken:** | 17.09.2022 |

## Welcome!

You made it to **Bayview Technologies, Inc.** premises! This exam is divided into **5 sections** each focusing on different areas. You will be answering questions with a total of **80 points**. You will be given **90 minutes** to complete the exam. It is very important to read each question and instruction carefully.

***Best of luck!***

## TEST CASE WRITING

|  |  |  |
| --- | --- | --- |
| **Instruction:** | Access Cashier page directly in [www.dafabet.com](http://www.dafabet.com)  Identify 10 test cases based on the Dafabet Cashier page and order each based on the priority.  **Pre-requisites:**  1. Create your own account via JOIN NOW button. User will be automatically logged in after successful registration.  2. Click CASHIER button. Each identified correct test case is worth **1 point** and each correct priority is worth **1 point**.  **Indicate registered account:**  Username: <username> Password: <password>  **NOTE:**  If www.dafabet.com is accessed as restricted page. Please do the following:  1. Download ModHeader as Chrome/Firefox extension (This is where you can modify HTTP request and response headers)  2. Add Profile / Request Header by clicking “+” or via menu.  Name: X-Forwarded-IP  Value: 111.90.158.20  3. Tick the checkbox of the newly added Request headers and click Resume Modheader icon before accessing the Dafabet page. |  |

| **TC#** | **Test Case** | **Priority** |
| --- | --- | --- |
| 1. **1** |  |  |
| 1. **2** |  |  |
| 1. **3** |  |  |
| 1. **4** |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## ANALYSIS

|  |  |  |
| --- | --- | --- |
| **Instruction:** | In Cashier page, find bug as much as you can and explain why you think it’s a bug. Each bug found is worth **1 point**. This is a **bonus** part! |  |

| **Bug#** | **It’s a bug because…** |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

## BUG REPORTING

|  |  |  |
| --- | --- | --- |
| **Instruction:** | Based on the bugs found in Part II, select a specific issue and file it as a bug report below. |  |

|  |  |
| --- | --- |
| **Issue Type: (1pt)** |  |
| **Bug Title: (3pts)** |  |
| **Severity: (3pts)** |  |
| **Priority: (3pts)** |  |
| **Pre-condition: (2pts)** |  |
| **Steps to Replicate: (2pts)** |  |
| **Expected Behavior: (2pts)** |  |
| **Actual Behavior: (2pts)** |  |
| **Environment: (2pts)** |  |

## PRIORITY DEFINITION

|  |  |  |
| --- | --- | --- |
| **Instruction:** | As a Quality Assurance Engineer, your main role is defining the priority.  For this section, you are expected to give the definition based on how you have done it in the past and give at least 1 bug sample per item.  Each Definition is worth **2 points** and each sample is worth **1 point**; you can get up to **5 points** each row. |  |

| **Priority** | **Definition** | **Bug Samples (1pt per item)** |
| --- | --- | --- |
| **1** | Low | Usually UI bug that does not block any UI functionality and is not disruptive from UX perspective. Eg. Wrong colouring or non centred DIVs. |
| **2** | Medium | Could be both front and Back End issue. Visible on the UI may affect UX. Eg. Avatar picture doesn’t load on users page. |
| **3** | High | Main functionality does not work affects UX but the work around exists. Eg. User cannot log in using 2FA but there’s possibility of login in using SSO. |
| **4** | Blocker | Main functionality does not work there’s not work around users are blocked from performing actions. Immediate fix is needed ASAP. Eg users cannot create accounts on the page or sign-up for new games. |

## ESSAY

|  |  |  |
| --- | --- | --- |
| **Instruction:** | Answer the questions provided below. Your answer must based on your own understanding or how you used it based on your experience.  You can get up to **5 points** for each question. |  |

1. Can you list down each development methodology you have experienced and give examples on how they fit your product delivery?

**I have worked with Agile and Waterfall methodology sometimes within the same project accordingly to the management needs at the moment. More often my role fit into Agile methods which included typical weekly and daily briefings all together with sprint planning and setting up the priorities. My role was to deliver the quality assurance of the sprint goals accordingly with the development progress. If necessary I could pick up on new items after discussing with my manger or team members. When the time allowed me I could focus on my personal projects eg. Automation or research for improving my QA skill set.**

1. Based on the projects that you have worked on, can you give examples on when Manual and Automation testing were applicable/necessary and why?

O**n early stage of production when the product or project was released the automation was usually not the top of my priorities since all the checks according to the documentation and test cases had to be done. When UI is firmly stable we could start automating the front end testing. If the project does not have the front end yet and only the services and their APIs are delivered the checkups can be performed alongside the initial automation processes eg. Using Postman or any other tool that can be attached to CI/CD pipelines. When UI is not delivered the only reasonable QA action is to organise endpoints into functioning workflows that give the picture of the future product/project and that will also shorten the testing time if for example the 80 endpoints have to be tested each day.**

1. What do you do after you execute test cases and did not find a bug?

**I can perform some black box testing based on my experience and see is if any bug was really missed. If everything seems correct and no braking issue was found I can pick up new task or start working on automating the test cases.**

**P.S**

**There will always be hidden bugs it is about acceptance criteria to verify we are good to go**

1. What is Smoke testing? Give an example on how you have used it on any projects that you have worked on.

Smoke testing is the type of test performed without test plan or test cases on the project that goes usually to live environment and was tested previously according to test plan. Smoke test does not include only testing the newly implemented functionality it also includes checks on another important modules and making sure nothing has broke on the process. Usually such a test should be performed by more senior testers that have higher understanding of the project/product.