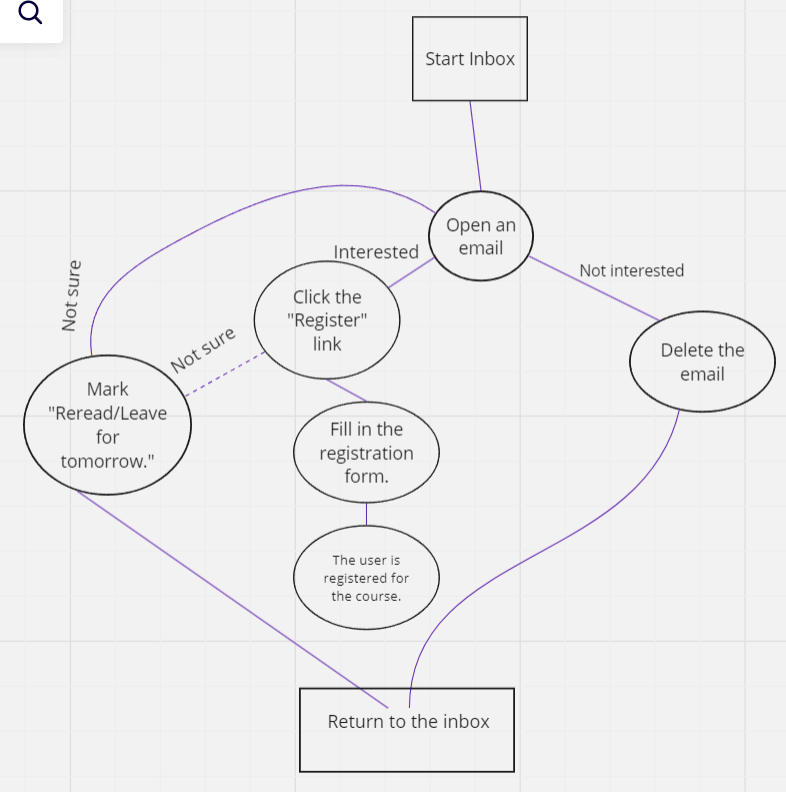
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Must have рівень:  1. Яке твердження щодо діаграми переходу станів і таблиці з тест-кейсами є вірним?     |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | Тест-кейс | 1 | 2 | 3 | 4 | 5 | | Стан початку | S1 | S2 | S2 | S3 | S3 | | Живлення | Power On | Power Off | RC On | RC off | Power Off | | Стан завершення | S2 | S1 | S3 | S2 | S1 |  1. Дані тест-кейси покривають валідні і невалідні переходи на діаграмі. 2. Дані тест-кейси показують всі можливі валідні переходи на діаграмі. 3. **Дані тест-кейси покривають деякі валідні переходи на діаграмі.** 4. Дані тест-кейси покривають пари переходів на діаграмі.   2. Співробітникам компанії виплачують бонуси за умови, що вони пропрацювати більше ніж рік та виконали заздалегідь поставлені цілі.  Ці умови можна подати у вигляді таблиці рішень:   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | Тест |  | Т1 | Т2 | Т3 | Т4 | | Умова 1: | Стаж більше року? | Так | Ні | Ні | Так | | Умова 2 | Ціль поставлена? | Ні | Ні | Так | Так | | Умова 3 | Ціль досягнута? | Ні | Ні | Так | Так | | Дія | Виплата бонуса | Ні | Ні | Ні | Так |   Який сценарій, що є ймовірним в реальному житті, пропущений в таблиці?   1. Умова 1 = ТАК, Умова 2 = НІ, Умова 3 = ТАК, Дія = НІ 2. Умова 1 = ТАК, Умова 2 = ТАК, Умова 3 = НІ, Дія = ТАК 3. Умова 1 = НІ, Умова 2 = НІ, Умова 3 = ТАК, Дія = НІ 4. **Умова 1 = НІ, Умова 2 = ТАК, Умова 3 = НІ, Дія = НІ** |
| Середній рівень:  1. Виконай завдання попереднього рівня.  2. Склади діаграму станів і переходів для тестування відеогри:  Після запуску гри ти опиняєшся в «Замку загадок», у кімнаті з двома коридорами — направо і наліво.  Якщо піти направо — потрапляєш до дракона. Дракон загадує тобі загадку. Якщо відповідь правильна — виходиш із замку та виграєш. Якщо неправильна — дракон загадує ще одну загадку. Якщо ти вдруге відповідаєш неправильно — дракон тебе з'їдає і ти програєш.  Якщо піти наліво — потрапиш до відьми. Відьма загадує загадку, якщо відповідь правильна — виходиш із замку і виграєш. Якщо відповідь неправильна — відьма переносить тебе до дракона.    3. Скільки тест-кейсів, відповідно до складеної діаграми, буде достатньо, щоб протестувати цю гру?  7 cases will be enough to cover the game. |
| Програма максимум:   1. Виконай завдання двох попередніх рівнів. 2. Продовжуємо розвивати стартап для застосунку, який дозволяє обмінюватися фотографіями котиків.   а. Напиши 5 use-кейсів для типової поведінки користувача твого застосунку.  б. Намалюй схему переходу станів на основі написаних користувацьких сценаріїв та склади таблицю рішень для одного з варіантів. |

|  |  |
| --- | --- |
| **Use Case 1** | Check the Offer Inbox |
| **Actor** | User |
| In the mornings, the user checks the Offer inbox. They sort the emails about courses they have there. Then they read the most interesting ones. They open each interesting email. They proceed with instructions to register for the courses. They delete any uninteresting emails with offers for courses. | |
| **Alternative Flow 1** | If they see an interesting offer, they proceed with links for registration. |
| **Alternative Flow 2** | If they decide to start the course later, they mark the email as “Reread.” |
| **Alternative Flow 3** | If they see nothing interesting, they delete the email. |

|  |  |
| --- | --- |
| **Use Case 2** | Check the Course Updates |
| **Actor** | User |
| Every day, the user checks the Course Updates. They sort the notifications due to their importance or priority. Then, they read the most important ones. They leave unread any non-urgent updates. | |
| **Alternative Flow 1** | If they see an important update, they proceed with clicking on it for further work. |
| **Alternative Flow 2** | If they decide to read the update later, they mark it “Non-urgent.” |
| **Alternative Flow 3** | If they see nothing crucial, they click the “Leave for tomorrow” button. |
| **Use Case 3** | Registering for the course |
| **Actor** | User |
| When the user checks the Offer Inbox and finds an interesting course they want to learn. They open the email. Then, they read the description and proceed to register. They mark with “Not interested” any course that they do not find appealing. | |
| **Alternative Flow 1** | If they see an interesting offer, they proceed with clicking on the “Register’ hyperlink for further work. |
| **Alternative Flow 2** | If they decide the offer is not interesting, they mark it as “Not interested.” |
| **Alternative Flow 3** | If they cannot decide on the course, they click the “Leave for tomorrow” button. |

User Case Scheme 3

<https://miro.com/app/board/uXjVMhDmlQ0=/?share_link_id=928782869329>



Decision Table Testing

The user opens the inbox and sorts the emails. If they are interested in the offered course in the email, they click on the register link. If they are not interested, they delete the email. If the user is not sure about the email offer, they mark it as "Reread." So there are three conditions: 1st condition: interested and 1 action: register. 2nd condition: not interested and 2nd action: delete. 3rd condition: not sure and action: mark as "Reread". When doing action 1, the user is redirected to the registration window; when doing action 2, the user is automatically redirected to the inbox; when doing action 3, the user returns to the inbox where they can see the marked email.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 |
| A. Interested | T | F | F | T | F | T |
| B. Not interested. | F | T | F | T | T | F |
| C. Not sure. | F | F | T | F | T | T |
| Action/Result | REGISTER | DELETE+INBOX | INBOX+ REREAD | X | X | INBOX+ REREAD |