

**DESIGN AND TESTING 2**

**SPRINT 3**

### 

|  |
| --- |
| **Group 2** |
| Guerrero Cuenca, Claudia |
| Macarro Klepsch, Miguel |
| Volante González, José Manuel |

|  |
| --- |
| **Content** |
| Current situation and retrospective of Sprint 3. |

**Content**

1. Finish level
2. Requirements and it’s owner
3. Retrospective
   1. Team retrospective
   2. Individual retrospective
4. Planning for the follow sprints

**1.- FINISH LEVEL**

* Deliverable 1 up to 10 points (including custom AssertJ)
* Deliverable 2 up to 10 points (including Cucumber UI tests)

**2.- REQUIREMENTS AND ITS OWNER**

**Requirement blocks:**

User stories have been distributed for each team member by implementation block. Each team member do the implementation and all test types about his blocks.

**Claudia Guerrero** 🡪 Implement a payment:

A secretary can do a payment of a visit. The payment can be pay with cash or creditcard.

**Miguel Macarro** 🡪 New system of making an appointment. New view with visits’ vet.

**Jose Manuel Volante** 🡪 Implement a diagnosis:

A vet can add a diagnosis to a visit. The diagnosis can have some prescriptions and medicines.

**Requirements implemented and testing in Sprint 3:**

The process that we have carried out in the sprint is:

1. Each team member has **implemented** the remaining functionalities of his implementation block. These consisted of the following user stories:

|  |  |
| --- | --- |
|  | **Done** |
| **Claudia Guerrero** | US18 |
| **Miguel Macarro** | US5a – US5b – US6 – US19 |
| **Jose Manuel Volante** | US17 – US20 – US21 |

We now have implemented 100% of the user stories.

1. All **unit tests** have been performed for each of the new methods created. We have done unit tests for controller, validators, services and query repositories. Positive and negative test has been done for each of these methods.
2. We created **parameterized tests** for some unit tests, among others, for example:
   * + - VisitValidatorTests
       - AdminControllerTests
       - CreditCardValidatorTests
       - PaymentValidatorTests
3. We added an **external service** to our application: The Lorem Flickr API. We use this API to obtain a random image for a given keyword and size via a GET request. We then show this image on our site.

1. We created **customized AssertionJ** assertions for all the classes in our model. We then replaced all uses of assertions in our unit tests with calls to the custom assertions where possible.
2. We created **24 UI tests**, one positive and one negative for each of 12 selected user stories.
3. Later, we transformed the UI tests to work with **cucumber**.
4. We created an **end-to-end test** suite for each **controller** in our application.
5. We created a **contract test** for the LoremFlickr **API** .
6. We created **database integration tests**.
7. Everything is in master. Then we do the **revision** of all things. Each member team has been assigned someone else’s tests to review and add new ones if necessary.

**Current situation**: 100 % of requirements for this Sprint are done.

**3.- RETROSPECTIVE**

**a.- Team retrospective**

Generally this Sprint went well because all members team work about the same and the tasks that proposed in the planning have been fulfilled.

With the implementation of new functionalities, we don’t have any problems.

We did have quite a few problems executing the UI test because the would unpredictably work some times and fail other times. After investigating the issue, we found out that it was caused by the chromedrive attempting to click on an element before it was actually loaded. We tried to solve this issue by introducing ChromeDriverWait statements but that didn’t work either. The solution that we finally came up with was to introduce Thread.sleep() statements. While not as elegant as using ChromeDriveWait, this solution makes that the tests work all the time.

The team has done three or four meetings all weeks. So if someone have a problem, the partners help with it in the meetings.

For this reason, nobody has been stuck in any task.

**b.- Individual retrospective**

Regarding individual perspective of each, it is believed that the Sprint went well and we don’t have any problem.

Individual performance in hours about this Sprint 3:

|  |  |
| --- | --- |
| **Member** | **Hours** |
| Claudia Guerrero | 75 |
| Miguel Macarro | 75 |
| Jose Manuel Volante | 65 |

**4.- PLANNING FOR THE FOLLOWING SPRINTS**

|  |  |  |
| --- | --- | --- |
| **SEMANA** | **PROPIETARIO** | **TRABAJO** |
| 2 marzo – 8 marzo | Miguel | Hacer correcciones proyecto base (Implementar cambio user 🡪 Person, Cambiar muchos Owner por 1), Implementar US1, US2 y US4 |
| Claudia | Implementar US7, US8 y US9 |
| Josema | Implementar US13, US14 y US15 |
| 9 marzo – 15 marzo | Miguel | Implementar US3, US5A, US5B |
| Claudia | Implementar US10, US11 y US12 |
| Josema | Implementar US16 y US17 |
| 16 marzo – 22 marzo | Miguel | Pruebas unitarias para correcciones, US1 – US5B |
| Claudia | Pruebas unitarias para US7 – US12 |
| Josema | Pruebas unitarias para US13 – US17 |
| 23 marzo – 29 marzo | Miguel | Automatización de pruebas con Travis |
| Claudia |
| Josema |
| **ENTREGA SPRINT 2** | | |
| 30 marzo – 5 abril | Miguel | Implementar US5a – US5b – US6  Implementar US19  Pruebas unitarias para US19 |
| Claudia | Implementar US18  Corregir error Vet 🡪 Owner  Pruebas unitarias para ello |
| Josema | Implementar US20 y US21  Inicio sin loguear  US17  Pruebas unitarias para US17 |
| 6 abril –  12 abril | Miguel | Pruebas unitarias para US5a – US5b – US6 |
| Claudia | Pruebas unitarias para US18 |
| Josema | Pruebas unitarias para US20 y US21 |
| 13 abril – 19 abril | Miguel | Pruebas de integración |
| Claudia |
| Josema |
| 20 abril – 26 abril | Miguel | Pruebas de integración  Pruebas end-to-end en los contoladores |
| Claudia |
| Josema |
| 27 abril –  3 mayo | Miguel | Pruebas de interfaz de usuario  (de lo implementado hasta el momento) |
| Claudia |
| Josema |
| 4 mayo – 10 mayo | Miguel | Pruebas de interfaz de usuario  (de lo implementado hasta el momento) |
| Claudia |
| Josema |
| **ENTREGA SPRINT 3** | | |
| 11 mayo – 17 mayo | Miguel | Pruebas de rendimiento |
| Claudia |
| Josema |
| 18 mayo – 24 mayo |  | Profiling del código |
| Claudia |
| Josema |
| 25 mayo – 31 mayo | Miguel | Refactorizaciones  Evidencias SonarCube |
| Claudia |
| Josema |
| 1 junio –  5 junio | Miguel | Proyecto A + |
| Claudia |
| Josema |
| **ENTREGA SPRINT 4** | | |