**Estimation**

|  |  |
| --- | --- |
| Sprint Backlog | Estimation |
| **Baseline Story**  As a HR Admin, I want to be able to create courses for Learners and trainers to enroll in.  **Acceptance Criteria:**   * I want to be able to create courses * I want to be able to set start and end dates for courses * I want to be able to set the prerequisites needed for learners to enrol for the courses * I want to be able to set the number of classes that can be opened under each course | **Estimation: 1**   * Create wireframe for course creation * Create API on Flask to insert new course data using information retrieved from course creation’s form into the database * Create form and required input fields to pass data through a form submission   **Explanation:**  We gave the baseline story a score of 1 point because our team felt that it is a good benchmark with the least difficulty and least effort. The webpage (Course creation page) can be done with minimal effort as it only consists of a form.  The function within the “Course Creation Page”, is simply to allow HR to fill in the details of a new course and it will be recorded into the database. |
| **1st User Story**  As an learner, I would like to do my quizzes in order to progress through each section  **Acceptance Criteria:**   * I am able to finish a course after passing the graded quiz created under it * I am able to access the quiz * I am able to view my mistakes for ungraded quizzes * I am able to see the number of past attempts on each quiz * I am able to retake graded quizzes until I hit the passing criteria | **Estimation: 3**   * Create a quiz-taking webpage for the learners to take the quiz * Database should included the quiz questions,options, timelimit and if the quiz is a graded quiz or not * Once the learner submitted the ungraded quiz, the webpage should show the learner’s answers and marked if it’s right or wrong.   **Explanation**  I gave the story a score of 3 as it takes a moderate amount of effort as compared to the baseline story. The story requires a several database tables to be created and referred to. The webpage (Do quiz page) also requires a form containing the questions and a countdown timer for the timelimit. The webpage (Quiz Results/attempts page) should include a clear way to show if the answers to the question is correct and all the previous attempts |
| **2nd User Story**  As a learner, I want to be able to enrol into courses that I am eligible for so that I can be a more competent engineer  **Acceptance Criteria:**   * I want to know if I am eligible to enroll in a specific course * I want to know view the time/schedule of class * I want to be able to view the course/class details (No of sections, Trainer attached) * I want to enrol for courses I am eligible for * I want to be able to view the prerequisites required for a course * I want to be able to view the start and ending enrolment dates * I want to be able to re-enrol in courses I failed | **Estimation: 5**   * Create a page with all the courses available and eligible for the learner to enrol * Database of enrolment, pending approval and prerequisite courses * The page will check if the user is eligible through checking the courses he/she has taken (API) * API should get all course details and using JavaScript/Vue, we can filter the course eligible. * Page should check if course is available to enrol (capacity, start and ending enrolment dates. * Once the learner enrolled to the course, the learner’s enrolment status will change to Pending. * If learner failed a course, the record in the database should not be restricting enrolment (Update to database is required)   **Explanation**  I gave this story a score of 5 as it seemed to be easy as it only contains the descriptions and filtering but there are details backend to prevent certain courses that are not eligible to be filtered. This takes more effort as compared to the baseline as many criteria are required to ensure smooth enrolment process for the learner. |