**Test Plan**

**Project: CiviHR Recruitment board**

Author: Mariana Guzun

Date: 24/02/2017

|  |  |
| --- | --- |
| **Test Objectives** | Test functionalities of CiviHR Recruitment Kanban board |
| **Project Overview** | Kanban screen is a significant improvement to the CiviHR recruitment module. The screen represents the visualisation of the candidates for one vacancy or job. Also it manages and tracks the applicant through the process. |
| **Requirements source** | 1. Brief requirements sent in a word document |
|  | 2. Wireframe screenshot |
| **Functionalities to be tested** | A list of questions on functionality that needs to be clarified with Product Owner can be found at [github link](https://github.com/mguzun/CiviHR-Task/blob/master/Questions%20for%20CiviHR%20Recruitment%20board.doc)  Test plan will have to be revised and adjusted after questions are discussed. |
|  | 1. Display of candidates in stages for one vacancy in Kanban board |
|  | 2. Filter candidate cards |
|  | 3. Filter vacancies |
|  | 4. Update candidate to new stage |
|  | 5. Enter transition comment when updating candidate stage |
|  | 6. Send email when updating candidate stage |
|  | 7. Add quick tasks when updating candidate stage |
|  | 8. Create task template in quick task |
|  | 9. Reject candidate – enter rejection comment and send email |
|  | 10. Add new candidate |
|  | 11. Edit Vacancy |
|  | 12. View only access for users with staff and manager permissions |
| **Functionality** | 1. Admin users can view, edit, add, reject in CiviHR Recruitment Kanban board |
| **Assumptions** | 2. Staff and managers cannot edit, add, reject in Kanban board, only view only permission |
| **Test Assumptions** | 1. Since CiviHR is a web-based app, assume testing is needed on the most frequently used browser which is Google Chrome. |
|  | 2. Testing will start after all open questions are clarified with Product owner |
| **Test Preparation** | 1. Consult CiviHR wiki to find more details on current Recruitment process |
|  | 2. Explore CiviHR demo hosting to understand current Recruitment process |
|  | 3. Identify user logins with Admin, staff, member permission |
|  | 4. Design functional test cases based on requirements |
|  | 5. Clarify all open questions with Business analyst and Product owner |
| **Test Environment** | Testing conducted on demo hosting site |
| **Test Platform** | Windows desktop with latest Google Chrome browser |

|  |  |
| --- | --- |
| **Testing types** | 1. Functional and UI testing: QA team manually executes planned test cases  2. Integration testing: manually execute test cases to verify integration between Recruitment module and Task module  3. Non-functional testing: the scope is to focus on functional requirements |
|  | 4. UAT: performed by end users and QA will provide their support on creating and execution of UAT script. |
|  |  |
| **Test scenarios** | Please find the list of test scenarios at [github link](https://github.com/mguzun/CiviHR-Task/blob/master/Test%20scenarios%20-%20CiviHR%20Recruitment%20board.xlsx) |
| **Test cycles** | 1. In QA Phase 1 - QA executes all planned tests |
|  | 2. In QA Phase 2 - test bugfixes, re-run all available tests to ensure no functionality regressed or new defects arise |
|  | 3. Next phases will be initiated if there are Critical and High defects found in previous phase that cannot be accepted for Go-live |
|  | 4. UAT Phase will be initiated after QA sign-off |
| **Test deliverables** | 1. Executed functional test script Enter link to executed script: |
|  | 2. All identified defects are stored in a word document until Approved for logging in defect tracking system Enter link to defects document: |
| **Test Estimates** | 10 m.d effort |