Acceptance tests

For ALL user stories in general:

Tested Requirement	Login in into a account
Quick Description	Once on the log in page: The user must provide email/username and password and press login button to sign in.
Prerequisite	1. Database for user accounts connected, configured and deployed. 2. Login request handling frontend and backend fully implemented.
Input	Email/Username, password
Expected Output	1. The console on the frontend must print successful login along with the user type, email and a Boolean set to true.
Observed Output	The user is redirected to the homepage according to the type of account.
Verdict	Passed
Comments	1. The login information for the user must be valid or it throws error. 2. The page must redirect to the homepage.

User story/employer

Tested Requirement	Create a job posting
Quick Description	Once on the job-posting creation page: The employer must be able to enter job title, employer name/company name, job description and press the “post” button to create a job posting.
Prerequisite	1. Database for job postings connected, configured and deployed. 2. The type of account must be ‘EMPLOYER’. 3. Job posting handling must be fully implemented
Input	Job title, Employer name/Company Name , job description
Expected Output	1. A new job posting must appear in the dataset. 2. In the backend console. A message indicating ‘insert posting’ must appear giving the structure and the data of the new job inserted.
Observed Output	1. A new job posting appeared in the dataset. 2. In the backend console, a message indicating ‘insert posting’ appeared giving the structure and the data of the new job inserted.
Verdict	Passed
Comments	Need to add a notification to inform the user in the frontend that it has been added for sprint 4.

Tested Requirement	Update a job posting
Quick Description	Once on the job-posting edit page: The employer must be able to edit job title, job position, company name, job description and press the “save changes” button to update a job posting.
Prerequisite	1. Database for job postings connected, configured and deployed. 2. The type of account must be ‘EMPLOYER’. 3. Job editing handling must be fully implemented
Input	Job title, Employer name/Company Name , job description
Expected Output	1. The job posting with updated inputs must appear in the dataset. 2. In the backend console, a message indicating ‘updating posting’ must appear giving the structure and the data of the job update.
Observed Output	Nothing. Update posting function needs to be rewritten.
Verdict	Failed
Comments	The implementation failed. The update function must be written. VERY IMPORTANT for sprint 4.

Tested Requirement	Delete a job posting
Quick Description	Once on the specific job-posting page: The employer must be able to delete the posting by clicking the “trash icon”.
Prerequisite	1. Database for job postings connected, configured and deployed. 2. The type of account must be ‘EMPLOYER’. 3. Job deletion handling must be fully implemented
Input	Click on the ‘trash’ icon.
Expected Output	1. The job posting must be deleted from the database. 2. In the backend console, a new prompt of ‘deleting posting’ must appear giving the structure and the data of the job update.
Observed Output	Nothing. Delete posting function needs to be rewritten.
Verdict	Failed
Comments	The implementation failed. The update function must be written. VERY IMPORTANT for sprint 4.

User story/student

Tested Requirement	Apply for a specific job posting
Quick Description	Once on the page the main page: The user of type 'STUDENT' must be able to apply for the job by clicking the 'apply' button.
Prerequisite	1. Database for job applications connected, configured and deployed. 2. Database for job postings connected, configured and deployed. 3. The user must be 'STUDENT' type account.
Input	Specific job post page, click to 'apply' button, employer account
Expected Output	1. The employer must receive the candidate's application. 2. In the database, for job applications, a new application must appear indicating which user applied and for which job applied.
Observed Output	Nothing happens. Need to rewrite the apply function.
Verdict	Failed
Comments	The implementation failed. The update function must be written. VERY IMPORTANT for sprint 4. Some user friendly features for sprint 4 maybe?: 1. A second button must appear that must display 'cancel application'. 2. The button must change to display 'applied' instead of 'apply'.

User story/Admin

Tested Requirement	See all students and employers profiles
Quick Description	Once on the homepage after logging in: The admin must be able to see all students and employers account.
Prerequisite	1. Database for user accounts connected, configured and deployed. 2. The user must be 'ADMIN' type account.
Input	A successful login page with type of account 'ADMIN'.
Expected Output	1. Display all the students and employers account 2. A remove button next to each employer and student.
Observed Output	Nothing. Admin has not implemented for sprint 3 as it was not a requirement for sprint 3.
Verdict	Failed
Comments	Not implemented yet. This is to be finished for sprint 4.

User story/Head Hunter

Tested Requirement	Refer candidates for jobs
Quick Description	Once on a specific job posting of an employer: The head hunter must be able to see all the applicants and click 'recommend' button to recommend students to employer.
Prerequisite	1. Database for user accounts connected, configured and deployed. 2. Database for job postings connected, configured and deployed 3. Database for job applications connected, configured and deployed. 4. The user must be 'HEADHUNTER' type account.
Input	Specific job post page, click to 'recommend button', head-hunter account
Expected Output	1. The button must display 'recommended'. 2. The employer must receive the recommendations of users. 3. In job posting application recommendation, the student id must be appended.
Observed Output	Nothing. Admin has not implemented for sprint 3 as it was not a requirement for sprint 3.
Verdict	Failed
Comments	Not implemented yet. This is to be finished for sprint 4.