



TECHNICAL DOCUMENTATION
OF
PLACEMENT
OPPORTUNITIES
WITH
ENTREPRENEURIAL
REACH
PLATFORM

The internal element of the technical documentation is the General summary of platform requirements, working file based on the conclusions of WP1



The report presents Placement Opportunities With Entrepreneurial Reach (POWER) platform with special focus on technical aspects. The platform is based and builds on the results from WP1 (interviews, surveys, analysis of other platforms etc.) and offers functional specifications of the POWER platform which are described in "General summary of platform requirements, Working file based on the conclusions of WP1". This document was used as a reference point during the development phase.

The report summarizes POWER platform version 1.0.

The repository of the software is stored in GitLab with detailed documentation of code and implemented changes.

The documentation is split into the following parts:

- Analysis of used technology, libraries and algorithms with special focus on its benefits.
- Platform structure
- Database structure
- List of MoSCoW classification features.
- Graphic diagrams of the application, visualising the architecture of the platform and adding Mockup designs.
- Flow of application (User Manual)



Index

| | |
|-------------------------------------------------------------------------|----|
| PART 1. Technology analysis for POWER Platform | 5 |
| 1. Docker | 5 |
| 2. PostgreSQL database..... | 6 |
| 3. Django framework..... | 6 |
| 4. Vue.js, Vuetify and Bootstrap..... | 7 |
| 5. Security issues..... | 8 |
| Cross-site request forgery..... | 8 |
| Encrypting the session..... | 8 |
| PART 2. Platform structure..... | 9 |
| PART 3. Database structure | 10 |
| PART 4. List of MoSCoW classification features..... | 11 |
| 1. Access to the platform / roles..... | 11 |
| Universities..... | 11 |
| Students..... | 12 |
| Incubators | 12 |
| Startups..... | 12 |
| Third parties..... | 13 |
| 2. Platform roles..... | 13 |
| 3. Core aspects summary..... | 15 |
| The core aspects covered in current version are as follows: | 15 |
| The core aspects not covered in current version are as follows: | 15 |
| Other aspects for consideration not implemented in current version..... | 15 |
| 4. Operational Requirements..... | 16 |
| 5. Design Requirements | 18 |
| 6. Functional Requirements..... | 19 |
| Authentication and Authorization | 19 |
| User Profile..... | 21 |
| Institutional profile (incubator) | 22 |
| Placement offers..... | 23 |
| Search engine | 24 |
| Alerts | 25 |



| | |
|-----------------------------------------------|-----------|
| Content management | 25 |
| Theming | 26 |
| Logs, Analytics and Monitoring..... | 27 |
| General requirements..... | 27 |
| 7. <i>Non-Functional Requirements</i> | 28 |
| PART 5. Application flow diagram | 31 |
| Appendix 1 Database structure | 33 |
| Appendix 2 Flow diagram | 34 |



PART 1.

Technology analysis for POWER Platform

Here we present the results of technology research we performed before we started development of the POWER platform. The main technology used are:

- Docker to create virtual environment to simplify further maintenance and management
- Vue.js

1. Docker

Docker is a set of Platform as a Service (PaaS) products that uses OS-level virtualization to deliver software in packages called containers. We deployed own Docker platform which is used to create new and robust instances of POWER platform containers. In each container the system or service is encapsulated and secured from external influence. Such an approach allows easy horizontal expansion of system in case of growing number of end users. The Docker technology is similar to VMware or VirtualBox, where the virtual machines are created but it has less limitations. Main advantages and limitations are presented below in table.

| Advantages | Disadvantages |
|-----------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------|
| It is possible to create separate containers with different functionalities. Flexibility of change. | All data must be stored in local disk using volumes. Backup of container is needed to protect database. |
| Small containers are easy to move and spread | One container is used to one functionality. There is need to use many small containers and manage them. |
| Best solution to protect all data | |
| Simple process to move whole platform | |
| Create virtual network which have influence to protect all system | |



| | |
|----------------------------------------------------------------------------------------|--|
| In Docker solution we mustn't install the operation system/ services might be deployed | |
| Docker use less hardware requirements like RAM, disk space, processors than other VM | |
| It is a faster than traditional solution based on virtual machines | |
| Open source | |

2. PostgreSQL database

PostgreSQL is an open-source, widely-used, reliable database which is used to store data about students, start-ups, applications and other information. The PostgreSQL database is implemented in one of the Docker containers. Main advantages and limitation are presented below

| Advantages | Disadvantages |
|----------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------|
| Open source | Installation and configuration are not as easy as a ready to use solutions. Needs preliminary configuration |
| Support ACID set of properties for database transactions (Atomicity, Consistency, Isolation, Durability) | |
| supports a variety of replication methods | |
| It is fastest database in open source area | |

3. Django framework

Django Framework is a Python-based free and open-source web framework that follows the model-template-view (MTV) architectural pattern. It allows easy implementation of API connector for communicating the view of platform with database. Django's primary goal is to ease the creation of complex, database-driven websites like the POWER platform is. The framework emphasizes reusability

and "pluggability" of components which will simplify further development. In meantime it requires less code and low coupling which leads to rapid development. Python is used throughout, even for settings files and data models. Django also provides an optional administrative create, read, update and delete (CRUD) interface that is generated dynamically through introspection and configured via admin models.

| Advantages | Disadvantages |
|----------------------------------------------------------------------------------------------------------------------------------|------------------------------|
| Designed to create application as fast as possible, effective and efficient | Use routing to specify URL's |
| Improve in solution security. Some of Internet treats are blocked and the main purpose is to use user authentication. | |
| This framework is better to use to scalable the application | |
| Versatile Content management, scientific computing platforms, and even big organizations, all these aspects are very efficiently | |

4. Vue.js, Vuetify and Bootstrap

Vue.js, Vuetify and Bootstrap frameworks are open-source Model–view–viewmodel JavaScript framework for building user interfaces and single-page applications. Vue.js features an incrementally adaptable architecture that focuses on declarative rendering and component composition. Advanced features required for complex applications such as routing, state management and build tooling are offered via officially maintained supporting libraries and packages.

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.

Vuetify is a Vue UI Library which extends its functionality. Similar to Bootstrap it contains many beautifully handcrafted Material Components which are used in final design. It extends HTML with HTML attributes called directives. The directives offer functionality to HTML applications, and come as either built-in or user defined directives.



| Advantages | Disadvantages |
|--------------------------------------------------------------------|-------------------------------------------------------------------------------------|
| Best Material Design Framework (it is leaning of Google solutions) | Documentation is dynamic and hard to use. Some functions are not well documented |
| provide clean, semantic and reusable components | |
| concerns style standardization and components reusability | |

5. Security issues

Security introduced in the POWER platform uses two basic mechanisms to secure the website.

Cross-site request forgery

Cross-site request forgery, also known as one-click attack or session riding (abbreviated as CSRF or XSRF), is a type of malicious exploit of a website where unauthorized commands are transmitted from a user that the web application trusts. There are many ways in which a malicious website can transmit such commands; specially-crafted image tags, hidden forms, and JavaScript XMLHttpRequests, for example, can all work without the user's interaction or even knowledge. Unlike cross-site scripting (XSS), which exploits the trust a user has for a particular site, CSRF exploits the trust that a site has in a user's browser. In a CSRF attack, an innocent end user is tricked by an attacker into submitting a web request that they did not intend. This may cause actions to be performed on the website that can include inadvertent client or server data leakage, change of session state, or manipulation of an end user's account.

One solution to avoid CSRF is to add an XCRF token to each form, which was implemented in POWER platform

Encrypting the session

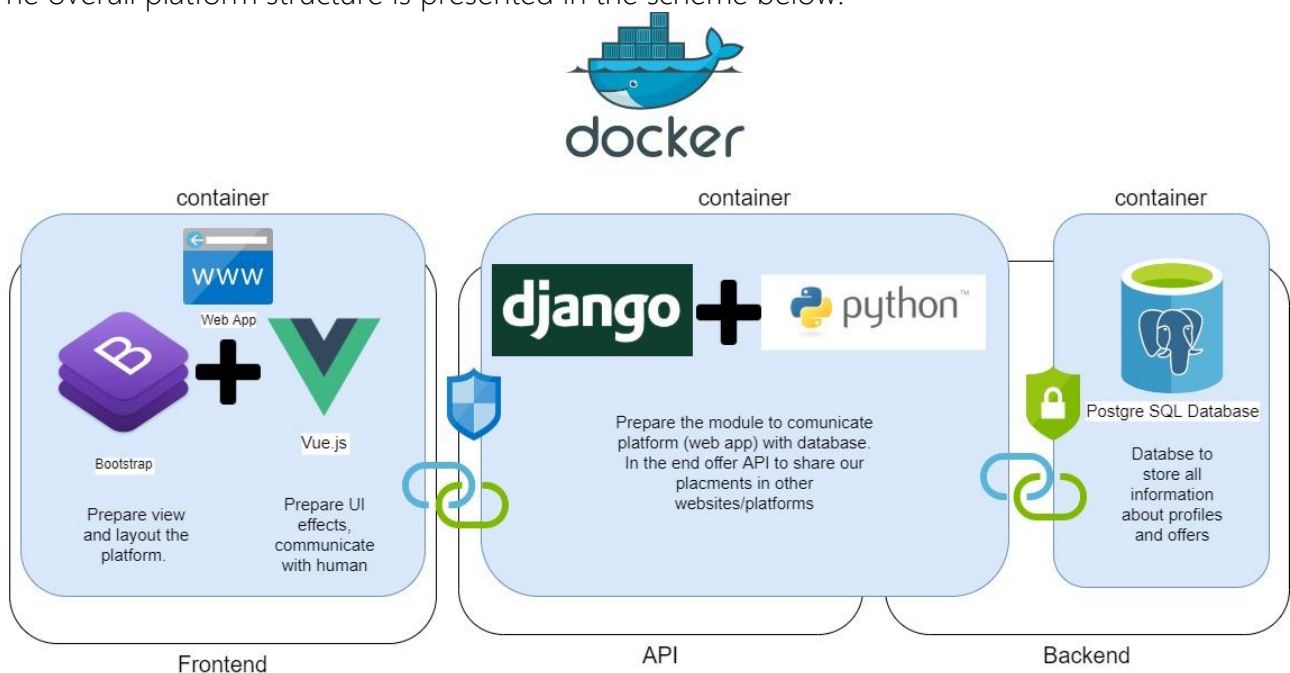
Each session is encrypted by HTTPS (Hypertext Transfer Protocol Secure). It is a secure version of the HTTP protocol, which task is to encrypt the information transmitted between the server and the client using the SSL / TLS protocol. As a result, thanks to HTTPS confidential data is secured and more difficult to intercept, read and change.



PART 2.

Platform structure

The overall platform structure is presented in the scheme below.



In the current version of platforms there are 3 docker containers:

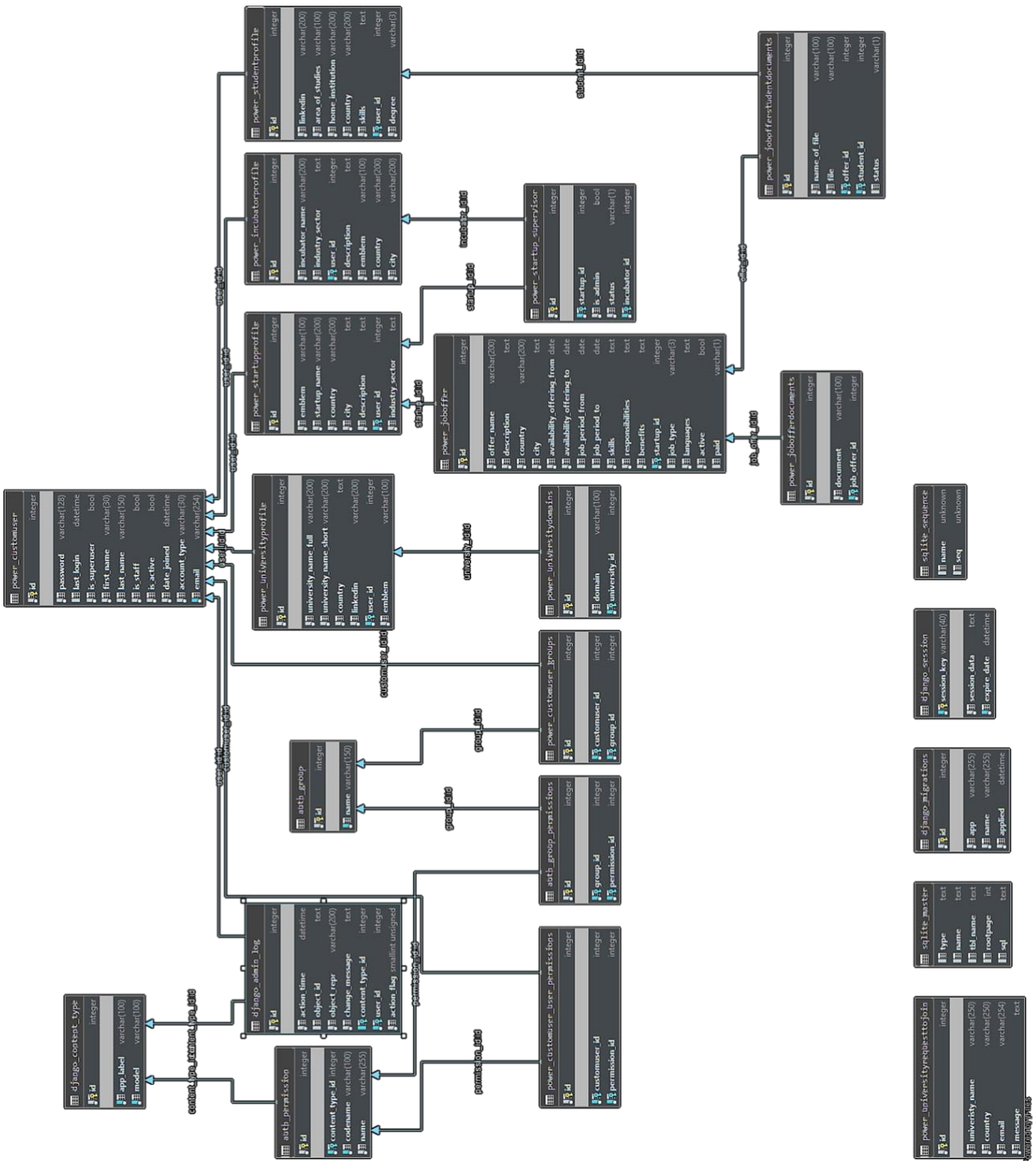
1. The front-end is realized using Bootstrap and Vue.js.
2. The backend is realized using Django and Python scripts.
3. The database is realized using PostgreSQL

The REST framework is used to communicate the frontend with the backend. The security issues are described in Part I.5.



PART 3. Database structure

Each field is defined in PostgreSQL database. The overall scheme with relation are presented in the graph below. Due to its size it is also an attachment to this documentation.





PART 4.

List of MoSCoW classification features.

1. Access to the platform / roles

The current version of platform is made to be available for:

- POWER incubators (and their respective startups)
- POWER universities (and their respective students)

There are following roles coded in current version of POWER platform:

- Student
- University
- Startup
- Incubator

Besides of them the system administrator role is implemented to added.

The splash screen allows to select the role and register to the platform. According to current assumptions the Incubators and Universities are predefined. Students can register themselves with one condition that their email belongs to the domains which are set by university.

Startup might register themselves but they need to be assigned to the predefined incubator.

Currently there is a single person representation for given account type.

The platform realizes the use cases described in General summary of platform requirements, Working file based on the conclusions of WP1 page 4-7.

Universities

Universities can be connected to POWER platform and they must set a contact person. The mechanism of recommendation and access request is implemented but not activated according the current project partners decision. The dashbord for university recommendation is not activated in current version.



Universities are able to share links with students to promote platform and single placement (including given placement offers).

Universities must define the email domains for students to limit their access to platform. Only students from defined domain will be able to register.

Students

Students might register themselves if only the allowed email domain is set by university. For example for Lodz University of Technology students are using official emails in domain edu.p.lodz.pl. This limitation allows to control the students' profiles and eliminate a fake accounts. During registration process a confirmation email is sent to activate account.

Each student is able to create its profile providing basic information about university, field of study, spoken languages, degree of study, etc. Moreover each student is requested to set up to 10 skills using predefined database. The same set is used for startups placement offers.

Student is able to search placement offers using filters and matching mechanism based on skills overlapping.

In current version the notification system is not implemented so the students have to follow new offers.

Each student can apply for given placement adding necessary predefined by startup documents.

Students can see list of applied placements offers.

Incubators

Incubators like universities can be connected to POWER platform and they must set a contact person. The mechanism of recommendation and access request is implemented but not activated according the current project partners decision. The dashboard for incubators recommendation is not activated in current version.

Each incubator can overtake the role of hosted startup.

In this version of platform incubators can browse only profiles of student who applied for given offer. Due to GDPR we limited the possibility of browsing all students profiles. This option is possible from technical point of view but it is evaluated by lawyers.

Startups

Startups can register and set up their profile. They can generate placement offers providing basic information like general placement description, duration, location, benefits, skills required, etc.



Moreover each startup placement offer's might require to add to student application up to 5 different attachments as a pdf files. After placement offer activation it becomes visible for student to be selected.

Startup can manage the application submitted by students and evaluate them in platform. Startups might look at students profiles who applies for give offer.

The role of startup might be executed by incubator.

Third parties

The API is not available in the current version but there is an easy mechanism to upload new offers to PostgreSQL database by platform administrators.

2. Platform roles

Platform users:

| | Registration and access | Core functionalities implemented in current version | List of functionalities, which are not implemented in current version |
|--------------|--------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Incubators | Full access after registration is completed | <ul style="list-style-type: none"> • can publish placement offers • can manage selection process on behalf of startup • can browse student profiles, who applied for given offer | <ul style="list-style-type: none"> • cant' browse student profiles (full access) – GDPR regulation • can't give specific extended access rights to affiliated startups |
| Startups | Access granted by respective incubator | <ul style="list-style-type: none"> • can access to student profiles who applied for given offer • manages student applications and selection process | <ul style="list-style-type: none"> • can't browse student profiles (full access) – GDPR regulation |
| Universities | Universities connect to the POWER platform via their own placement | <ul style="list-style-type: none"> • share information with students • promote single placement by link sharing | <ul style="list-style-type: none"> • Dashboard of statistics and data is not implmented/shown |



| | | | |
|----------|-----------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--|
| | management platform (to ensure placement offer visibility on the local platforms) | | |
| Students | Via registration on the platform | <ul style="list-style-type: none"> • can view and browse POWER placement offers through platform • can search for relevant offers by country, field and skills • can apply for specific placement(s) of their interest • have an option to provide extended information about their profile and save it on the platform • student is identified by email domain | |

3. Core aspects summary

The core aspects covered in current version are as follows:

- Registration mechanism and essential information from both sides (students-startups)
- Notification system once there is a match (student-startup) to provide relevant information and guidance about crucial steps both to students and startups
- QA assurance surveys and tracking mechanism
- Ongoing support during the placement that takes place both virtually (via materials) and face-to-face (by support on the spot)
- Displaying relevant profiles/offers for registered users
- Startup request and possibility to add extra pdf files to student application

The core aspects not covered in current version are as follows:

- Notification sent to startups after the placement offer expires (2 weeks) - to confirm the status and whether the startup will host an intern (or not)
- Deletion of the profiles after certain period of time (both startup offers when they expire and student profiles, when they graduate) – is done manually upon request.

Other aspects for consideration not implemented in current version

- Statistics and information tracking (for universities, about their students and their satisfaction)
- Statistics for incubators (number of views, student and startup satisfaction level)
- Certificates at the end of the placement
- Contracts with startups (sample templates provided)
- Alerts for both sides (if matching profiles appear in the platform)
- Event calendar (at incubators)
- Indication of the status of startups (looking for an intern - in the selection process - open for spontaneous applications) and students (looking for placement opportunities - in the selection process - selected)



4. Operational Requirements

The operational requirements and its status of implementation are summarized in a table below:

| ID | Requirement | Description | Remarks to current version |
|-----|--------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------|
| OR1 | Support diverse users | Must support students, incubators and startups, also HEIs (with various ICT skills) | Done |
| OR2 | Be accessible to all the identified target groups/users | The platform needs to provide access to all the target groups & stakeholders, providing the same usability and user experience, regardless the location and the device they use to access the platform (PCs, tablets, smartphones). | Done, Platform supports RWD |
| OR3 | Be operational | Must be operational through the whole year | Done, system is permanently monitored, actions are stored in log |
| OR4 | Support different roles and permissions for users | As users can be of different types (e.g. students, incubators, startups), different user roles must be supported | Done |
| OR5 | Be able to integrate with Existing Institutional Services and other relevant services (like, other platforms at the expansion stage) | The system must be able to integrate with/connect to existing services of HEIs (existing placement platforms, placement listings, etc.). Also, since there are already relevant services (erasmusintern.org, for instance), it would be beneficial to the platform to connect with such external services (during the expansion stage). | Not implemented |



| | | | |
|-----|-------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------|
| | | The system must be able to create connections with existing services of HEIs – in-house (at respective HEIs) or third-party systems (external). | |
| OR6 | Support customization of user permissions | As the user permissions are related to the type of target group (student, incubator, startup), the permissions given to the specific user role must be configurable to the needed level (in particular between incubator and startup). | Done, incubator might play role of startup |
| OR7 | Be able to extend the user groups with minimum effort | New users (incubators, universities) should be able to connect to the platform with minimal effort. The platform implementation must support the expansion of user institutions and the architecture must be adequate to support a large number of additional users. | Done, not activated |
| OR8 | Support Multilingual Content | As it is foreseen to have a broad audience within the multicultural and multilingual environment of EU in the future, the platform must support multilingual content. At the first development phase of the platform it is required to support English and the official languages of the project partners. However, English is expected to be the core language of the platform. | Done, Each student can add information in any language (i.e. in pdf files) All menus are provided in English |



5. Design Requirements

The design requirements and its status of implementation are summarized in a table below:

| ID | Requirement | Description | Remarks to current version |
|-----|----------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| DR1 | User support | The platform requires the participation of people with different skills-set, including software engineers, university/faculty staff members, incubator representatives and students, therefore the flow of communication must be efficient, considering the objectives of the platform. | Done, platform is easy to use |
| DR2 | Compliance with GDPR | The platform must be compliant with the GDPR that came in force since the 25th of May 2018, in terms of the required legislative texts, the required mechanisms to support users edit, export or delete their data and the required security measures to ensure personal data leakage. | Done, Startups and incubators can browse only profiles of students who applied for given placement offer. Database is protected and maintained by authorized administrators. |
| DR3 | Support agile methodologies | The implementation must follow incremental activities where base functionalities are implemented and tested in advance of extra functionalities. The agile methodologies must provide several releases of the platform in a timeframe, including the correction of malfunctions from the previous releases. | Done GitLab and Trello is used during development. |
| DR4 | Develop on top of open source software | Where possible, use open source software. Additionally, contribute to | Done |



| | | | |
|-----|----------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------|
| | | open source projects where this is beneficial for the project community. Moreover, where possible, release platform specific custom software components source code as open source after consideration with the project Steering Committee. | We base on open-source, free technologies described in Part 1 |
| DR5 | Support functional testing | Functionalities can only be considered as supported when their implementation in software components has been tested and the results are aligned with the expected output. | Done, |

6. Functional Requirements

Authentication and Authorization

| ID | Requirement | Description | Priority | Current version status |
|-----|---------------------------------------------|----------------------------------------------------------------------------------------------------------------|----------|------------------------------------------------------------------|
| FR1 | Users | Management of Users | MUST | Done |
| FR2 | User registration through Registration Form | Support username/password registration | MUST | Done |
| FR3 | CAPTCHA on user Registration Form | The registration form should implement the latest CAPTCHA technology to prevent a number of security breaches. | MUST | No. Students must use email domains. Bots are not allowed. |



| | | | | |
|------|----------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------|---------------------|
| FR4 | Login with Credentials through Login Form | Support login with username (e-mail) and password. | MUST | Done |
| FR5 | Support expiration mechanisms of authentication | Support expiration of authentication tokens to prevent security issues. | MUST | Done, 24h |
| FR6 | Email confirmation on standard registration procedure | Confirm a user registration only after email confirmation. A confirmation email should be valid for a short period of time (e.g. 24 hours) | MUST | Done |
| FR7 | Password recovery via email | Users must be able to recover a lost or compromised password by using their registered email. | MUST | Done |
| FR8 | Support policies to renew credentials | Support policies that require the renewal of passwords after a certain period (e.g. each 12 months), requires password to be different than the previous one. If a user does not respond to the password renewal policy, put the account on stall, and delete after some reasonable time. | COULD | Done |
| FR9 | Support account creation by incubator (invitation based) | Incubator is pre-filling the required information to acquire access rights. | MUST | Done, not activated |
| FR10 | Login and register via EduGAIN | Support Login and Register with an EduGAIN Account. Acquire the required user data (name, photo, | COULD | Not available |



| | | | | |
|------|------------------------------------------------------------------|-------------------------------------------------------|--------|----------------|
| | | etc.) after authorization from the user. | | |
| FR11 | Support Dual-factor authentication for sys-admin and admin-staff | Increase security levels of sys-admin and admin-staff | SHOULD | Not available, |

User Profile

| ID | Requirement | Description | Priority | Current version status |
|------|----------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------|----------|------------------------|
| FR12 | User Profile | Users can have their profile with basic info and contact details. Users are able to create, update, or delete a user profile. | MUST | Done |
| FR13 | User profile Image | Ability to upload, crop, delete, replace a user image. A default avatar must be present in case a user has not provide one. | MUST | Done |
| FR14 | Favorite placement offers (for students) or student profiles (for startups/incubators) | Users can save (and remove if no longer relevant) placement offers and student profiles to their favorite list | SHOULD | Done |
| FR15 | Favorite Search Queries | Users can save and remove search queries on their favorite search list. | SHOULD | Not available |
| FR16 | Placement offer Alerts | Users can create alerts based on the favorite search queries they have defined. An e-mail or other | SHOULD | Not available |



| | | | | |
|------|------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------|--------|---------------------------------|
| | | notification may be created upon match. | | |
| FR17 | Notifications about student applications | Users (startups/incubators) should receive notifications when student applies for a placement | SHOULD | Done |
| FR18 | Users can share and upload info as deemed relevant (more detailed CV and/or additional info about startup) | In addition to mandatory fields, users are able to upload additional information to complete their respective profiles. | MUST | Done Only pdf files |
| FR19 | Export User Profile Data | As required by GDPR, a user must be able to export all the data related with the user. | MUST | Done, as sys-admin |
| FR20 | Account Settings | Users can have access and edit all the details of their accounts (GDPR). | MUST | Done |
| FR21 | Delete Account | Users can delete their accounts and all their info on the platform (GDPR). | MUST | Done, by sys-admin upon request |
| FR22 | Choice of pre-defined info (areas of studies, skills, location etc.) | Select the fields only from a pre-defined list (to enable matching functions). | MUST | Done |

Institutional profile (incubator)

| ID | Requirement | Description | Priority | Current version status |
|----|-------------|-------------|----------|------------------------|
|----|-------------|-------------|----------|------------------------|



| | | | | |
|------|---------------------------------------------------------------------------------------------|------------------------------------------------------------|--------|----------------------------------------------------------------------------------|
| FR23 | Institution's basic information | Display institution's basic information | MUST | Done |
| FR25 | Institutions services and related startups | Display institution's services and related startups | SHOULD | Done |
| FR26 | Startups that are open placement hosting | Display startups that are open placement hosting | MUST | Done |
| FR27 | Display incubators (and start-ups) by region/country | Users can filter startups by region/country | MUST | Not implemented, Each incubator has access to all startups which belongs to them |
| FR28 | Create, edit and delete institution's profile | Can create, edit and/or delete its profile on the platform | MUST | Done |
| FR29 | Incubator can add their respective startups to the platform (giving relevant access rights) | Can manage respective startup profiles | MUST | Done |

Placement offers

| ID | Requirement | Description | Priority | Current version status |
|------|-----------------------------------------------|------------------------------------------|----------|------------------------|
| FR30 | Placement offer - basic information displayed | Display basic information | MUST | Done |
| FR31 | Link to the application form | Display a link to the application form | MUST | Done |
| FR32 | Other (related) placements | Display other (related) placement offers | SHOULD | Not implemented |



| | | | | |
|------|------------------------------------------|---------------------------------------------------------------------------------|------|-----------------|
| | | | | |
| FR33 | Add to Favorites | Users can add a placement offers to their favorites or remove it from this list | MUST | Not implemented |
| FR34 | Create, Edit and Delete placement offers | Owners of the placement offer can create, edit and delete it | MUST | Done |

Search engine

(incubators/startups search for student profiles and students search for placement offers)

| ID | Requirement | Description | Priority | Current version status |
|------|--------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------|----------|------------------------|
| FR35 | Placement offers basic search | A search bar is enabled where users can enter keywords and search for relevant placement offers | MUST | Done |
| FR36 | Search by pre-defined matching functions (advanced search) | Searching enabled via specific categories (skills, fields, study areas etc.) | MUST | Done |
| FR37 | Search hints | Provide hints to search based on the frequently searched keywords | SHOULD | Not implemented |
| FR39 | Incubator/startup can view and download student applications | Incubators/startups can generate a pdf file with the student application profile | MUST | Done |
| FR40 | Create connection/match | Students and startups should notify the platform/system when there is a match (to be able to follow next steps) | MUST | Done |
| | Provide/guide to | Ensure access to the QA surveys | MUST | Not implemented |



| | | | | |
|------|---------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------|------|-----------------|
| FR41 | expectation survey | (including via university and incubator assistance) | | |
| FR42 | Access to statistics and satisfaction level | Universities and incubators should have access to the statistics of placements, as well as results of the surveys (expectations/satisfaction level) | MUST | Not implemented |

Alerts

Alerts mechanism is not implemented in current version

| ID | Requirement | Description | Priority |
|------|------------------------------------------------------------------|---------------------------------------------------------------------------------------------|----------|
| FR43 | Notification on new placement offers & relevant student profiles | Periodic email notification when new offers or student profiles are available | COULD |
| FR44 | Create notifications based on the user's saved search query | When a user saves a favorite query, he could be able to enable notifications on this query. | SHOULD |
| FR45 | Create notifications based on the user's profile | A user should be able to enable notifications on the info provided on his/her profile | COULD |
| FR46 | User defined period of notifications | User could define the period of the notifications (daily, weekly, bi-weekly, monthly) | SHOULD |

Content management

Content management concept is changed. Each profile might be edited according the needs using Markdown editor. The system administrator can modify the below elements.

| ID | Requirement | Description | Priority |
|----|-------------|-------------|----------|
|----|-------------|-------------|----------|



| | | | |
|------|-------------------------------|--------------------------------------------------------------------------------------------------------------------|------|
| FR47 | Pages | Management of pages | MUST |
| FR48 | Menus | Management of menus | MUST |
| FR49 | Image Content | Support images in pages with different formats, including (jpeg, png, gif, svg) and the possibility to add images. | MUST |
| FR50 | Video Content and file upload | Support videos in pages and file uploads | MUST |
| FR51 | Access Control | Support the access to information according to defined policies and roles | MUST |
| FR52 | Publish/Unpublish content | Publish and unpublish content (pages, placement offers etc.) | MUST |

Theming

| ID | Requirement | Description | Priority | |
|------|-----------------------------------------------------|-------------------------------------------------------------------------------------------------------------|----------|------------------------------|
| FR53 | Reorder Menus | Support customization of menus (some parts of them) | COULD | See CMR (FR47-FR52) |
| FR54 | Responsive Design | Users will be able to visit the platform from all devices (desktops, tablets, smartphones) | MUST | Done Bootstrap, Vue.js |
| FR55 | Aggregate Statistics (in welcome page or elsewhere) | There should be shown basic statistics as number of placements, count of students and/or startup companies) | SHOULD | Done, not activated |



| | | | | |
|--|--|--|--|--|
| | | | | |
|--|--|--|--|--|

Logs, Analytics and Monitoring

| ID | Requirement | Description | Priority | |
|------|-------------|---------------------------------------------------------------------------------------------------------------|----------|------|
| FR56 | Debug Log | The platform must store information about Errors, Warnings and notices. | MUST | Done |
| FR57 | Error Log | The platform must store diagnostic information and error recording that it encounters in processing requests. | MUST | Done |
| FR58 | Access Log | The platform must store information about all requests processed by the server. | MUST | done |

General requirements

| ID | Requirement | Description | Priority | |
|------|----------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|----------|------|
| FR59 | Contact and Suggestions form | A contact and suggestions form must be present. The person receiving the responses must be the admin-staff and/or an admin-staff configurable mail list. | SHOULD | Done |
| FR60 | Page to show Website is in maintenance | A page to show that the website is in maintenance must be shown, where an update or maintenance occurs. | MUST | done |



| | | | | |
|------|------------------------------------|----------------------------------------------------------------------------------------------|--------|-------------------|
| FR61 | Link and webpage errors (i.e. 404) | Errors pages must be included and presented when an HTTP coded error occurs. | MUST | done |
| FR62 | Practical information | A page presenting practical information on the host country or institution could be present. | SHOULD | Under preparation |

7. Non-Functional Requirements

The non-functional requirements include requirements that ideally must be supported by all the components of the platform and are summarized below.

| | | | |
|------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|--------------------------------------|
| Auditability | Must register all user actions and services to allow system audits | MUST | Done |
| Scalability | The platform must have the ability to add capacity (and users) to a deployed system over time. Scalability typically involves adding resources to the system but should not require changes to the deployment architecture. | MUST | Done Provided by Dockers |
| Security | The platform must ensure secure authentication and authorization of users, as well as the secure transport of information. | MUST | Done, HTTPS tokens |
| Usability | The platform must provide efficient and easy to use interfaces. | MUST | Done, to be checked with users |
| Interoperability | The platform should provide well-defined APIs in order to easily integrate with other platforms and | MUST | Not implemented |



| | | | |
|---------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|-------------------------------------|
| | services. | | yet. Manual upload data is possible |
| Configurability | The platform must support certain levels of customization, which include the options in menus, theming, logos, among others. | MUST | Done, only sys-admin |
| Flexible deployment | The platform must support different deployment options, which can include cloud-based deployment, distributed deployments using diverse virtual machines or containerization. | MUST | Done, Provided by Docker |
| Adaptability | The platform and the various components must adapt to different conditions, in terms of hardware, environment, number of users to support. | MUST | Done, Provided by Docker |
| Availability | The platforms' resources and services must be accessible to end users | MUST | Done, Provided by Docker |
| Extensibility | The design and the implementation of the platform must have the ability to extend with the minimum required effort. | MUST | Done, Provided by Docker |
| Fault Tolerance | The platform must be able to continue operating properly in the event of the failure of some of its components. | MUST | Done, Provided by Docker |
| Maintainability | The platform must be easy to maintain. | MUST | Done, Provided by Docker Dev-ops |



| | | | |
|---------------|-------------------------------------------------------------------------------------|------|----------------------------------|
| | | | environment is under development |
| Integrability | The platform must be able to integrate the discrete sub-components into one system. | MUST | done |



PART 5.

Application flow diagram

The application flow diagram is realized in MS Visio. The .vsdx file is attachment no 2 to this documentation.



Co-funded by the
Erasmus+ Programme
of the European Union





Appendix 1

Database structure



Appendix 2

Flow diagram

Flow diagram was done in Microsoft Visio. The pdf version was added.