Co-funded by
the European Union

Learning Record Store (LRS)

R3 Documentation and User Guideline

Manual for End Users

Guide for Institutions and Authorities

Technical Handbook for Developers

This document was created as part of the ERASMUS+ project "Partial certification in the vocational field of event technician - PACE-VET ", project ID: 2021-1-DE02-KA220-VET-000028156.

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Education and Culture Executive Agency (EACEA). Neither the European Union nor EACEA can be held responsible for them."

Project manager

Betriebswirtschaftliches Forschungszentrum für Fragen der mittelständischen Wirtschaft e.V.
Contact: Jörgen Eimecke
info@bfm-bayreuth.de
https://www.bfm-bayreuth.de

Project partners

VPLT – Der Verband für Medien- und Veranstaltungstechnik e.V.
Contact: Guntars Almanis
info@vplt.org
https://www.vplt.org/

Erasmus Hogeschool Brussel
Contact: Johan van den Broek
Johan.van.den.broek@ehb.be
https://www.erasmushogeschool.be/nl

Steunpunt voor de produktionele, ontwerpende, en technische krachten van de brede culturele sector
Contact: Chris van Goethem
chris@stepp.be
http://www.stepp.be

Overleg Scholing Arbeidsmarkt Theatretechniek
Contact: Els Wiljmans
info@osat.nl
https://www.osat.nl/

Table of contents

1. Introduction

By providing an exhaustive documentation alongside the application and the requirement specification, this PR supports the easy replicability by outside parties and an easier to follow replication of the validation process in other sectors of the economy.

Especially the documentation belonging to the application (Technical Handbook) will facilitate developers with an interest to pick up the PR 1 and 2 of the PACE-VET project to build on existing groundwork, thus lowering the threshold of transferring the process to other sectors and enabling a broader and sustainable usage of the final product.

Addressing the different target groups, you will find below:

the Manual for End Users (Professionals in event technology)

the Guide for Institutions and Authorities

the Technical Handbook for Developers

as well as information on quality management and an outlook for the PACE-VET app.

The appendix contains the following in English, German, and Dutch:

- Quick Guide for Registration (short guide to the PACE-VET app and registration) an
- iOS Guide (short guide for registration for Apple devices).

Further information and video tutorials, as well as the source code for the PACE-VET app, can be found on the PACE-VET project website (https://pace-vet.eu/app/).

2. Manual for End Users

2.1 Target audience: Event technician level 4

This manual provides information and instructions on how to use the app's various functions and options. The PACE-VET app is an innovative application designed to enable technicians working in the live performance and event industry to document and validate their acquired competences in a transparent way.

PACE-VET provides a digital infrastructure that serves as a basis for the recognition and validation of competences of live performance and event technicians, thus enabling better adaptation of vocational education and training to the needs of the labor market and users. All knowledge, skills and/or competences that a person has acquired and/or can demonstrate after completing a formal, non-formal or informal learning process are recognized. It enables a reliable procedure for the validation of prior learning (VPL) and facilitates lifelong learning. PACE-VET also aims to create a pathway to a certification process for acquired competences of technicians by developing a digital platform to collect acquired certificates with the help of microcredits.

It serves as a personal database to map acquired competences and enables an EU-wide transfer of information about candidates and their acquired certificates. A backbone for acceptance within the EU is the direct connection to ESCO[1]: the European multilingual classification of skills, competences and occupations.

Eventually, the app is intended to support self-directed learning processes and provide access to learning services, personal development opportunities and labor markets. Currently, PACE-VET offers a limited group of competence units at EQF level 4 of a basic technician in the live performance and event industry.

The European live performance and event industry faces several sector-specific circumstances that present unique challenges for employees and employers to find each other. There are two main reasons for this: Innovation and mobility. Firstly, the live performance and event industry, which is both subject to and driving technological change, is highly innovative and job skills are therefore inherently associated with an extremely short lifespan. This is particularly true for technicians. Technicians are often required to update their skills immediately after obtaining their qualification. In many cases, learning takes place informally - "on the job". These constant cycles of updating knowledge and skills on the job lead to the current situation where employees and the self-employed acquire multiple competences over the course of their working lives that are neither formally certified nor validated. Sometimes they are not even recognized by an employer or employee as essential

[1] ESCO: European Skills, Competences, Qualifications and Occupations

skills for a particular job. As a result, event technicians often struggle to demonstrate their mastery of the latest technology.

Employers, in turn, find it difficult to ensure the suitability of their staff for certain types of jobs. The live performance and event technology workforce is also highly mobile and often works on international projects or for foreign employers. Due to the different understanding of job profiles in the European Member States, matching workers' competences with employers' expectations is a major obstacle in the employment process.

2.2 The Portfolio System

Structured portfolios enable the documentation and consequently the assessment of learners' competences against learning outcomes aligned to specific unit standards. The digital portfolio plays an important role in PACE-VET as it contains the results of different assessment methods and evidence of learning experiences linked to corresponding (groups of) competences to document the learner's individual skills in application in an objective way. Due to the diversity of evidence, particular attention must be paid to reliability. PACE-VET defines the specific forms of evidence that can be accepted. Candidates own their portfolios and access to them.

The structured portfolio offers candidates the opportunity to record and disseminate the competences, skills and knowledge they have acquired. It can also serve as an assessment tool (self-evaluation).

Forms of evidence to be used in the portfolio:

- **Artefact/product:**

Where competences and skills require candidates to produce an artefact or physical product, the artefact or product must be made available to the mentor and assessor.

The learner must provide evidence of the following:

  - Details of the tasks given to learners that correspond to the assessment criteria of the units concerned,
  - A declaration by learners that all work they produce is their own,
  - Summative, learner-generated assessment evidence - teaching material must not be used as evidence.

- **Recorded activity/practical ability:**

Evidence must be provided that the candidate is individually and actively completing tasks that demonstrate that they fulfil the assessment criteria. Evidence may be assessed by direct observation of performance and must consist of at least two of the following:

  - Labelled images,

- Detailed witness statements,
- Video (with narrative or written protocol),
- Logbook/assessment of the learner,
- Peer observation reports.

N.B.: When using images/videos, each individual learner must be clearly identifiable.

2.3 The Units and Competences

Currently, PACE-VET is using the results of the Erasmus+ project "TeBeVAT"[2] to define specific "units", which are groups of competences that can be assessed with the recognised assessment methods. They can also be considered as "micro-diplomas" because, to quote a recent Cedefop research paper, "They offer certain advantages over traditional qualifications, notably their greater flexibility and their suitability for building sector or occupation-specific competences (retraining/up-skilling) to respond to changing industry needs. The definitional boundaries between micro-qualifications and sectoral or occupational credentials are blurred; quality-assured and industry-recognised certificates can be considered as a sub-category of micro-qualifications that enjoy greater visibility, recognition and trust."[3]

Currently, two units are offered in the occupational profiles for "Lighting" and "Sound". In the future, the application should offer the full range of learning outcomes for a live performance and event technician on an EQF level 4 within the sector as described in the units of the TeBeVat project:

- Lighting (ready for use)
- Sound (ready for use)
- Mechanical equipment (in preparation)
- Energy distribution (in preparation)
- Environment stage (in preparation)
- Video and media integration (in preparation)
- Work organisation unit (in preparation)

The PACE-VET process aims to strengthen the holistic educational approach of VET qualifications with low-threshold access but can also support adult education in order to accelerate the learning process.

PACE-VET:

- promotes lifelong learning,

[2] TeBeVat: Validation of informal learning in the professional field of event technicians.

[3] Cedefop, Microcredentials for labour market education and training - Microcredentials and evolving qualifications systems, Publications Office of the European Union, 2023, https://data.europa.eu/doi/10.2801/566352, last checked on 10 Januaryth , 2023

- enables the recognition of prior learning experiences, especially informal and non-formal learning,
- provides a path for continuous learning to acquire and improve skills and competences, and
- enables partial certification of competences in this area.

2.4 Functionalities of the app

The PACE-VET app is a web app that can be accessed from any device, is cloud-based and offers security and data protection. Tools and accessibility are defined by the different user roles.

Annex 1 contains an overview of the functions for technicians. Other users have this manual for reference. For help with setting up the app on an iPhone, see annex 2.

2.5 The Process

After the candidate has created his portfolio and if needed consulted the mentor, the documents on file are checked.

- Proof of valuation:

It is important that the evidence for the assessment is determined individually. Awarding organisations and centres offering these assessments must also meet the assessment and quality assurance requirements of the process.

- Certificates and qualifications:

It is important that the three levels of quality assurance in relation to certification quality - Input/Process/Output - are taken into account when trying to assess and validate the certifications offered. Qualifications should be checked against the Cedefop (European Centre for the Development of Vocational Training) definition of certification, which refers to "individuals achieving learning outcomes that 'meet' certain standards and/or requirements". Therefore, learning outcomes-based standards should be a key element in the certification process.

Learning outcomes are "statements of what a learner knows, understands and is able to do when he has completed a learning process; they are defined in terms of knowledge, skills and competences".

- Witness statement/peer evaluation:

When accrediting prior learning, the assessor may not be able to observe how the candidate performs certain aspects of their work. In this case, it may be appropriate for another person to comment on the performance by completing a statement known as a 'witness statement' or peer evaluation. Witness statements and peer evaluations should only be used to support other evidence such as a product. Requirements are:

  - They must be submitted by a person who has no family tries or friendship bonds to the applicant and is able to give a valid opinion on their performance, e.g. a supervisor, line manager, customer or client,
  - Contain comments that relate specifically to the performance criteria,
  - They must be authenticated by the signature of the witness, his/her function, address, telephone number and date.
- Contracts and worksheets:
  - The evidence must be recent. How long ago was this proof provided and what does it say about the mastery of the process?
  - Is the evidence verifiable?
- Requirements for the forms of evidence
  - Authentic: It must be clear that the evidence has actually been carried out or achieved by the candidate or is related to the candidate. The experience must come from activities that have been carried out independently or in groups and in which the candidate has made a significant contribution to the results.
  - Relevant: related to the competence to be assessed. The experience must have been gained in relation to the function of the job profile. The candidate must state which tasks and activities he/she has carried out and the results of these activities. He/she must also state why they are relevant to the activities carried out.
  - Sufficient level: The evidence must reflect the level of competence expected for the qualification or certificate.
  - Topicality: still have value in a current working environment.
  - Quantitative: The evidence must be of sufficient scope and supported by sufficient experience (time spent exercising a competence).
  - Diverse: Concretising the breadth and scope of experience. Preferably the candidate presents evidence from different "angles" (not one-sided).

Candidates can apply for an assessment or publicise their portfolio to get in touch with potential clients.

2.6 The Mentor

A mentor provides guidance and counselling to prepare a candidate for the assessment process and the recognition of prior learning. The mentor helps the candidate to set clear goals for their personal development and gives advice on further education, learning or access to the labor market. Mentors only become active when a candidate gives them permission to do so.

Mentors can:

- See the profile of the candidates,
- See the evidence associated with the candidate,
- Coaching candidates, giving feedback and writing assessment reports,
- Propose valuation options,
- Support the candidates in this process.

2.7 The Assessor

An assessor measures a candidate's existing competences, skills and knowledge against a specific qualification or certificate according to a predetermined standard or procedure and objectively assesses whether the candidate meets the relevant performance criteria.

assessors can:

- Assess candidates = measurement of whether a candidate possesses the competences required in a profile or set of competences according to standardized procedures,
- Validate evidence or assessment results.

2.8 The Employer

Employers who have been given access to the profiles of applicants can check whether these technicians have the right skills to fulfil a task or fill a vacancy.

Employers can:

- Search for technicians on the basis of:
  - Responsibilities
  - Region
  - Language

2.9 The Certifying Body

PACE-VET requires a central authority at European level for the recognition of certification bodies and assessment centres (see 4.1).

Certification bodies | Assessment Centres can:

- Review and certificate the assessment results,
- Provide digital credentials (e.g. European Digital Credential {EDC} - Europass)

Certification bodies | Assessment Centres must:

- Comply with the quality standards defined by ETTEC - based on the ISO-IEC_17024:2012 standard = General requirements for bodies that certify persons.
- Provide digital certificates as proof of assessment in the required PACE-VET format,
- Provide information and schedules for assessment opportunities.

2.10 Quality Control

Assessment Centres:

They are certified by ETTEC and must comply with the ISO-IEC_17024:2012 standard.

Assessor:

They are certified by ETTEC and ensure that the examination centre has the necessary equipment and facilities for the intended examination methods.

Rating:

At least two assessors are involved in the assessment to increase validity and ensure equality and fairness in the validation process. Validity, reliability and authenticity are increased through the use of various methods of authentication. Although these methods are time-consuming for the candidate, they allow evidence of skills to be recognised in an objective way using different assessment tools.

In PACE-VET, learning outcomes should be validated by the results of at least three assessment methods (triangulation). The portfolio should always be a part of the assessment process. Due to the virtually unlimited possibilities of AI, it is of utmost importance that the evidence is not based on only one source and that the sources can be verified by the assessors.

2.11 Limitations of the LRS

Due to the project's limited resources, the application and the Learning Record Store (LRS) are currently restricted to basic functions. In the future, however, the aim is to create a genuine semantic, open learning model for technicians in the live performance and event industry that facilitates all self-directed learning processes.

At present, the LRS should offer the following:

- A lifelong learning documentation tool that includes a structured portfolio and an applicant profile with CV / and related documents on the applicant's ability to apply knowledge and skills independently and responsibly,
- Information on current competences and occupational profiles in the sector based on ESCO and the sectoral level being developed as part of the TeBeVat project,

- Information on the respective competence units, which are summarised in assessable micro-credit points,
- Access to:
  - Mentors
  - Assessment Centres
  - ETTEC - for information and participation in appointment procedures
- Information on possible validation and certification procedures,
- Options to enable access to information:
  - Mentors
  - Evaluation centres and assesors
  - Employers

Future LRS services will be explained in more detail in section 6.

2.12 Data security, Governance and Ownership

All information uploaded by the candidate is confidential and belongs to the candidate. No information will be disclosed to other parties without the express consent of the candidate. The candidate must be informed in detail of the reason for sharing the information with other parties and must verify the release of information or data.

3. Guide for Institutions and Authorities

There are currently no tools in the PACE-VET application to include input from educational institutions and authorities, due to the challenges of quality management.

When defining the competences and designing the certification and the quality standard, the consortium drew on the results of the international non-profit organisation ETTEC (European Council for Qualification and Certification of Stage and Event Technicians), as this certification and recognition process has already proven its worth in several European countries.

ETTEC's mission is the mutual recognition of certificates and qualifications in the field of live performance and entertainment.

ETTEC is the consolidation of the work on certification, qualification, job description, skills and quality that has been developed and carried out by organisations and individuals across Europe over the last 15 years. Most of this work has been supported by the European Commission through various mobility and partnership programmes.

ETTEC is based on the results and findings of the following EU projects:

- **TTT-LPT** 2005 - 2009, development of the concepts for describing competences on a lowest common denominator basis
- **Cue-one-Go** 2011 - 2012, local project from Flanders that demonstrates the principles of self-assessment and assessment of dual learning
- **ESCO** 2013 - 2015 ESCO, development of a Europe-wide taxonomy for skills, competences and occupations
- **ETTE** 2014 - 2017, development of a standard and an assessment procedure for a basic safety certificate and initial development of the assessment profile
- **TALQ** 2015 - 2017, definition of the principles of joint recognition based on ESCO
- **TeBeVat** 2018, development of portfolio assessment, development of mentor profiles.
- **LAAR** 2017 - 2019, integration of the use of a learning folder in management systems.

At the beginning of 2020, the core partners of these projects came together and founded the ETTEC organisation.

Ideally, the app could offer the following: (see 4)

- Opportunities to use the LRS in VET-education
- transferability – from the school system to the LRS
- feedback through the app
- assessment through the app
- further development of the LRS - adding plug ins

4. Technical Handbook

Introduction

The PACE-VET platform has been developed to facilitate the certification and validation of vocational competences across the EU by using modern web technologies and cloud infrastructure. It provides a robust and scalable solution for the management of digital micro-credentials, enabling the secure transfer and verification of workers' skills and qualifications. The system includes both a mobile application developed with Flutter and a backend infrastructure based on the Phoenix Framework, PostgreSQL and containerised services in cloud environments such as Fly.io.

This technical manual is designed to provide developers and contributors with a comprehensive guide to setting up, running and maintaining both the app and the backend services of PACE-VET. It covers everything from initial setup to deployment, including containerisation, database management and important considerations for integrating external services such as S3-compatible storage. Our approach combines ease of development with a scalable architecture that can handle high demand while ensuring reliability and security.

4.1 PACE-VET Flutter Application

4.1.1 Overview

This handbook provides step-by-step instructions on setting up, running, and maintaining the PACE-VET Flutter application. The application includes features such as API interaction, internationalization (i18n), routing, and ORM generation. Additionally, it provides support for error tracking via Sentry and generating release builds for Android and iOS.

4.1.2 Project Setup

4.1.2.1 Environment Configuration

Create a `.env` file at the root of the project to configure the API URL and Sentry DSN. The `.env` file should have the following structure:

```
{
        "API_URL": "URL-TO-API-ENDPOINT",
        "SENTRY_DSN": ""
}
```

- *API_URL*: The endpoint URL for the backend API.

- *SENTRY_DSN:* The Data Source Name (DSN) for Sentry, used to track and log events and errors.

4.1.2.2 Installing Dependencies

To install all the necessary Flutter dependencies, run the following command:

```
flutter pub get
```

4.1.3 Generating Source Code

4.1.3.1 Generating Translation Files

The PACE-VET app uses Flutter’s localization system to manage translations. To generate the translation files, run the following

```
flutter gen-l10n
```

This will generate the localized content from files under the ‘lib/l10n‘ directory.

4.1.3.2 Generating Code (Routing, ORM, etc.)

To generate source code related to routing, ORM, and other parts of the system, use *build_runner:*

```
flutter pub run build_runner build
```

Alternatively, you can use the following commands to simplify setup:

```
flutter pub get

flutter gen-l10n

flutter pub run build_runner build
```

4.1.3.3 Fix for Build Process Bug

There is a known bug in the build process for one of the dependencies. To fix it, manually create the necessary file using the following command:

```
echo ’name:’ > .dart_tool/flutter_gen/pubspec.yaml
```

4.1.3.4 First-Time Setup with Visual Studio Code

If you’re using Visual Studio Code, you can automate the first-time setup by running the "first time setup" task, which will execute all the necessary commands after *‘flutter pub get‘.*

4.1.4 Running Tests

4.1.4.1 Unit Tests

To run unit tests for the project, use the following command:

```
flutter test
```

4.1.4.2 Integration Tests

For integration tests, which test the app’s functionality as a whole, use the following:

```
flutter␣test␣./integration_test
```

4.1.5 Localization and Translations

4.1.5.1 Generating Translation Files

To generate the translation files, you can run:

```
flutter gen-l10n
```

This will compile localization files found under the *‘lib/l10n‘ directory*.

4.1.5.2 Updating Translations with *Lyrebird*

To update and manage translations, run the following command to start *Lyrebird*, a tool for managing translations:

```
flutter␣pub␣run␣lyrebird
```

This will launch *Lyrebird*, where you can review and update the translations.

4.1.6 Running the App with *Sentry*

To run the app with *Sentry* enabled for error tracking, use the following command, replacing <DSN> with your actual Sentry DSN:

```
flutter run--dart-define=SENTRY_DSN=<DSN>
```

When run with the correct DSN, *Sentry* will begin tracking and logging events, helping you to monitor and diagnose issues within the application.

4.1.7 Generating a Release Build

4.1.7.1 For Android

To create a production-ready build of your Android app, use one of the following commands:

```
flutter build appbundle--release--dart-define=API_URL=https://pace-vet.fly.dev/
```

or

```
flutter build apk--release--dart-define=API_URL=https://pace-vet.fly.dev/
```

These commands will generate an optimized *app bundle (.aab)* or *APK (.apk)* for pro duction, configured with the specified API URL.

4.1.7.1.1 Configuring Internet Permission for Release Builds

To enable internet access in release builds, ensure that the ‘INTERNET‘ permission is specified in your ‘*AndroidManifest.xml*‘ file.

Open the manifest file in your Flutter project:

android/app/src/main/AndroidManifest.xml

Within the ‘‘ *tag* and before the ‘‘ *tag*, add the following per mission:

Adding ‘‘ *ensures* network access in release builds. While internet permissions are typically granted by default in debug builds, they must be explicitly declared in release mode to enable internet connectivity.

4.1.7.1.2 Custom Android Configuration

The *android/app/build.gradle* file includes custom configurations specific to this project. Review and adjust these settings based on your project’s requirements and development environment.

- Plugins Section

In addition to standard plugins, this project includes a custom *Flutter Gradle plugin*:

```
plugins {

id "com.android.application"

id "kotlin-android"

id "dev.flutter.flutter-gradle-plugin"

}
```

- Namespace

A custom namespace is defined for the application:

```
android {

namespace "be.ehb.pace_vet"
}
```

- Java Compatibility Options

Java compatibility is configured to use Java 1.8:

```
compileOptions␣{
␣sourceCompatibility␣JavaVersion.VERSION_1_8
␣targetCompatibility␣JavaVersion.VERSION_1_8 }
```

- Signing Configurations

Custom signing configurations are set for both release and debug builds. Important: Replace *your_password and /path/to/your/key.jks* with actual keystore details. Avoid committing sensitive information to version control.

```
signingConfigs {
    release {
        keyAlias 'key'
        keyPassword 'your_password'
        storeFile file('/path/to/your/key.jks')
        storePassword 'your_password'
    }
    debug {
        keyAlias 'key'
        keyPassword 'your_password'
        storeFile file('/path/to/your/key.jks')
        storePassword 'your_password'
        }
    }
```

- Dependencies

The project includes additional dependencies for error-prone annotations and Java an notations:

```
dependencies {

        implementation ’com.google.errorprone:error_prone_annotations:2.7.1’

        implementation ’javax.annotation:javax.annotation-api:1.3.2’

}
```

These configurations ensure the project meets specific environment requirements and enable features such as error handling and custom build signing.

4.1.7.2 For iOS

This section provides step-by-step instructions for generating a production-ready release build of the app for iOS using Flutter. This release build will be configured to connect to the production API endpoint.

4.1.7.2.1 Prerequisites

1. Apple Developer Account: Ensure you have an active Apple Developer account to manage certificates, provisioning profiles, and app distribution on the App Store.

2. Mac Computer with Xcode Installed: iOS builds require Xcode, which is only available on macOS. Ensure you’re using the latest version of Xcode.

3. Flutter Setup: Make sure Flutter is installed and properly configured for iOS development.

4.1.7.2.2 Steps to Generate a Release Build

For this release build, set the production API endpoint. Use ‘*--dart-define*‘ to specify the API URL. This will ensure the app connects to the correct server in production:

```
flutter build ios--release--dart-define=API_URL=https://pace-vet.fly.dev/
```

After running the Flutter build command, open the iOS project in Xcode to configure signing and additional settings.

The ‘*--dart-define=API_URL=...*‘ allows for different API endpoints in development and production. Make sure to replace ‘*https://pace-vet.fly.dev/*‘ with the correct pro duction URL.

Generating a release for iOS involves building the Flutter project with the production API URL, configuring code signing in Xcode, archiving the app, and distributing it via App Store Connect. This ensures that your app is ready for deployment on the App Store, with all required configurations for production.

4.1.8 Summary of Commands

Here’s a consolidated list of the most important commands for setup, testing, and release:

Project Setup

```
flutter pub get

flutter gen-l10n

flutter pub run build_runner build
```

Fix Build Bug

```
echo ’name:’ > .dart_tool/flutter_gen/pubspec.yaml
```

Running Tests

```
# Unit tests

flutter test

# Integration tests

flutter test ./integration_test
```

Localization and Translation

```
flutter gen-l10n

flutter pub run lyrebird
```

Running with Sentry

```
flutter run--dart-define=SENTRY_DSN=<DSN>
```

Generating Android Release

```
flutter␣build␣apk␣--dart-define=SENTRY_DSN=␣--dart-define=APP_ENV=production
```

4.1.9 Conclusion

This chapter provides all the necessary steps to configure, build, and maintain the PACE VETFlutter application. By following the outlined steps, you can set up the development environment, generate the necessary source code, and build the application for release, while also integrating testing and error monitoring using Sentry.

4.2 PACE-VET backend

4.2.1 Introduction

This handbook provides a step-by-step guide for setting up, running, and deploying the PACE-VET Phoenix application. The application is built using the Phoenix framework and integrates with PostgreSQL and S3-compatible storage. Additionally, deployment and scaling are handled using Docker/Podman and the Fly.io platform.

4.2.2 Setting Up the Phoenix Server

4.2.2.1 Ensure PostgreSQL is Running Locally

Before running the PACE-VET application, ensure that a PostgreSQL server is running locally or within a container. If you prefer to run it in a container, refer to the "Running in Containers" section below.

4.2.2.2 Install and Set Up Dependencies

Run the following command to install all necessary dependencies and set up the database:

mix␣setup

4.2.2.3 Start the Phoenix Server

You can start the Phoenix server using one of the following methods:

- Standard mode:

bash mix phx.server

- Inside an interactive Elixir shell (IEx):

iex-S mix phx.server

4.2.2.4 Access the Application

Once the Phoenix server is running, you can access the application at:

http://localhost:4000

4.2.3 Running in Containers

For containerized environments, you can run PostgreSQL and S3 locally using Docker or Podman. This ensures that your application dependencies are isolated and easy to manage.

4.2.3.1 PostgreSQL in Docker/Podman

To run the PostgreSQL database in a Docker or Podman container, use the following command:

- Docker:

```
docker run--name pacevet-postgres

-e POSTGRES_PASSWORD=postgres-p 5432:5432-d postgres
```

- Podman:

```
podman run--name pacevet-postgres

-e POSTGRES_PASSWORD=postgres-p 5432:5432-d docker.io/postgres
```

4.2.3.2 S3-Compatible Storage with MinIO

To set up a local S3-compatible storage server, you can use **MinIO**. The following Podman command starts the MinIO container:

```
podman run-d--name pacevet-s3

-p 9000:9000-p 9001:9001 quay.io/minio/minio server /data--console-address ":9001"
```

You can then access the MinIO web console at:

http://localhost:9001

Login credentials for the web console:

- Username: minioadmin
- Password: minioadmi

4.2.4 Running the Application in Containers

4.2.4.1 Generate the Application Image

First, you need to build the application image using Podman (or Docker). Run the following command:

```
podman build--tag pace_vet .
```

4.2.4.2 Run the Application Container

After generating the image, start the application container with the necessary environ ment variables, such as the *SECRET_KEY_BASE* and *DATABASE_URL*:

```
podman run-e SECRET_KEY_BASE=<keybase>

-e DATABASE_URL=ecto://<user>:<password>@<hostname>/<database>
```

Replace the placeholders:

- *<keybase>*: The secret key base for your Phoenix application.
- *<user>*: Database user.
- *<password>*: Database password.
- *<hostname>*: Hostname of the database.
- *<database>*: Database name.

4.2.5 Deploying with Fly.io

4.2.5.1 Connect to Fly.io

Application To connect to the Fly.io infrastructure and interact with the deployed app, follow these steps:

4.2.5.1.1 Generate a WireGuard configuration for your Fly application

fly wireguard create

4.2.5.1.2 Move the generated configuration to your system

mv /etc/wireguard/fly.conf

4.2.5.1.3 Bring up the WireGuard connection

wg-quick up fly

4.2.5.2 Access Remote IEx Console

Once the WireGuard connection is established, you can access the remote IEx console by running:

./observer.sh

This command connects the local IEx console to the remote Phoenix node running on Fly.io.

4.2.6 Connecting to the Remote PostgreSQL Database

To proxy the Fly.io-hosted PostgreSQL database to your local machine for direct access, run the following command:

fly proxy 54322:5432-a pace-vet-db

This command proxies the remote database port to your local machine, allowing you to access it via port `*54322*`.

4.2.7 Releasing the Application

4.2.7.1 Deploy to Fly.io

To release a new version of the PACE-VET application, use the Fly.io deployment command:

```
fly deploy
```

4.2.7.2 Running Database Migrations and Seeds

If the database has been reset or updated, you'll need to SSH into the application to run migrations and seed data:

SSH into the application using the Fly.io console:

```
bash fly ssh console--pty-C "/app/bin/pace_vet remote"
```

Once connected, run the following commands to apply migrations and seed the database:

```
elixir PaceVet.Release.migrate PaceVet.Release.seed
```

4.2.8 Managing the S3 Storage

If you need to reset the environment or clear the S3 storage, you can do this by SSH-ing into the Fly.io console and running the following command:

```
elixir PaceVet.Release.clear_s3
```

This command clears out the connected S3 storage, allowing you to reset the environment if needed.

4.2.9 Summary of Key Commands

Local Setup

```
bash mix setup mix phx.server iex-S mix phx.server
```

Containers

###- PostgreSQL in Docker/Podman:

```
bash docker run--name pacevet-postgres

-e POSTGRES_PASSWORD=postgres-p 5432:5432-d postgres
```

###- S3 with MinIO:

```
bash podman run-d--name pacevet-s3-p 9000:9000

-p 9001:9001 quay.io/minio/minio server /data--console-address ":9001"
```

Application Deployment

Generate Image:

```
podman build--tag pace_vet .
```

###- Run Application:

```
podman run-e SECRET_KEY_BASE=<keybase>
-e DATABASE_URL=ecto://<user>:<password>@<hostname>/ <database>
```

Fly.io Deployment

###- Deploy:

```
fly deploy
```

###- Run Migrations:

```
elixir PaceVet.Release.migrate PaceVet.Release.seed
```

4.2.10 Conclusion

This handbook provides all necessary instructions to install, run, and deploy the PACE VET Phoenix application, whether locally or in a containerized environment. With the use of PostgreSQL, S3-compatible storage, and Fly.io for deployment, the application is built with scalability and flexibility in mind.

4.3 Open sources

The software used for the PACE-VET app and all written materials, such as this guide, are Open Educational Resources (OER) in the sense of "Creative Commons". They may be used free of charge, but the project partners must be notified if they are passed on. Reproduction and storage in electronic systems is permitted after notification. However, the content of the database with the current profiles of the candidates is not OER due to data protection regulations.
The source code can be found here: https://pace-vet.eu/app/

5. Quality management

5.1 Quality standards

The project partners see ETTEC (European Council for Qualification and Certification of Stage and Event Technicians) as the European organisation that can provide, monitor and update the quality standards necessary for the mutual recognition of the TeBeVAT and PACE-VET procedures in the member states. ETTEC is an international non-profit organisation that promotes the international mutual recognition of certificates in the field of event technology and performing arts in Europe. It can certify national assessment centres and certification bodies to ensure mutual quality, develop common professional profiles and train assessors, mentors and trainers. It adheres to EN ISO/IEC 17024: 2012-11 - Conformity assessment - General requirements for bodies certifying persons. The standardised assessment procedures and qualification requirements for assessors in the PACE-VET process contribute to quality assurance.

PACE-VET requires a central authority at European level for the recognition of certification bodies and assessment centres (see 4.1). The entire procedure must be disseminated in the sector, among VET certification bodies and on the labour market.

To ensure the credibility of both the certificates and the provider, a single authority for the sector is required to certify the validity of the certificates issued.

The following components ensure compliance with quality management throughout the entire process:

- Code of Conduct
- Procedures and standards
- Training and validation
- Agreed and validated content
- Rules and paths to equal opportunities
- Appeal procedure

All organisations involved in the process must act accordingly (in accordance with ETTEC):

- Please note the data protection guidelines (GDPR),
- Observe the transparency guidelines for processes and documents,
- Ensuring the professionalism of the personnel deployed,
- guidelines for the utilisation of external service providers (outsourcing),
- Creation of guidelines for people with special needs,
- Define your organisational structure and the distribution of responsibilities,
- Ensuring quality assurance through internal and external measures,
- Definition of a procedure for the submission and processing of complaints and objections (e.g. ombudsman).

5.2 Process of procedure evaluation

Candidates who have undergone an assessment procedure by an approved and accredited assessment centre may discuss the procedure or assessment results with the centre and, where appropriate, with the mentor(s) and representatives of the centre.
Candidates can view and review the assessment procedures. The assessment results are documented and are available to the candidates. The candidate has the right to request an appeal procedure.

5.3 Objection procedure

Candidates can view and review the assessment procedures. The assessment results are documented and are available to the candidates. The results can be contested by:

- First contact with a coordinator as part of a mediation process,
- If necessary, by appealing against the assessment result via an ombudsman to a complaints body at national or international level

6. Future areas of application of the LRS tool

In the future, the LRS could provide the following services:

- Expand data collection capabilities to document all learning experiences: Quantifiable, shareable and trackable activities related to learning outcomes,
- Creation of a personal development plan based on the applicant's learning outcomes and goals,
- Access to information about training and learning opportunities,
- Opportunities for self-assessment,
- Information on new essential and optional skills and qualifications in the sector,
- More specific units/microcredentials such as special effects/pyro, rigging, sound specialist level 5 etc,
- Extension to all 24 official languages of the EU, starting with the three "procedural languages" - English, French and German,
- Constantly updated job offers that match the applicant's profile,
- Further expansion of links to job opportunities and employment possibilities
- Option for applicants to set up user groups within the application to exchange information and experiences, tips and pitfalls about the app or even within the sector.

7. Conclusion

This comprehensive documentation contains all the important information to ensure easy reproducibility by external parties. The developed principles allow easy access to the process and thus an easier replication of the validation process for other economic sectors, which ensures a more sustainable use.