

<https://doi.org/10.1038/s41524-025-01936-4>

# Making atomistic materials calculations accessible with the AiIDAlab Quantum ESPRESSO app

Check for updates

Xing Wang<sup>1,2,10</sup>✉, Edan Bainglass<sup>1,2,10</sup>, Miki Bonacci<sup>1,2,10</sup>, Andres Ortega-Guerrero<sup>3,10</sup>, Lorenzo Bastonero<sup>4</sup>, Marnik Bercx<sup>1,2</sup>, Pietro Bonfà<sup>5,6</sup>, Roberto De Renzi<sup>7</sup>, Dou Du<sup>8</sup>, Peter N. O. Gillespie<sup>6</sup>, Michael A. Hernández-Bertrán<sup>5,6</sup>, Daniel Hollas<sup>9</sup>, Sebastiaan P. Huber<sup>8</sup>, Elisa Molinari<sup>5,6</sup>, Ifeanyi J. Onuorah<sup>7</sup>, Nataliya Paulish<sup>1,2</sup>, Deborah Prezzi<sup>6</sup>, Junfeng Qiao<sup>8</sup>, Timo Reents<sup>1,2</sup>, Christopher J. Sewell<sup>8</sup>, Iurii Timrov<sup>1,2</sup>, Aliksandr V. Yakutovich<sup>3</sup>, Jusong Yu<sup>1,2</sup>, Nicola Marzari<sup>1,2,4,8</sup>, Carlo A. Pignedoli<sup>3</sup>✉ & Giovanni Pizzi<sup>1,2,8</sup>✉

Despite the wide availability of density functional theory (DFT) codes, their adoption by the broader materials science community remains limited due to challenges such as software installation, input preparation, high-performance computing setup, and output analysis. To overcome these barriers, we introduce the Quantum ESPRESSO app, an intuitive, web-based platform built on AiIDAlab that integrates user-friendly graphical interfaces with automated DFT workflows. The app employs a modular Input-Process-Output model and a plugin-based architecture, providing predefined computational protocols, automated error handling, and interactive results visualization. We demonstrate the app's capabilities through plugins for electronic band structures, projected density of states, phonon, infrared/Raman, X-ray and muon spectroscopies, Hubbard parameters (DFT+ $U+V$ ), Wannier functions, and post-processing tools. By extending the FAIR principles to simulations, workflows, and analyses, the app enhances the accessibility and reproducibility of advanced DFT calculations and provides a general template to interface with other first-principles calculation codes.

Density-functional theory (DFT) simulations play a pivotal role in condensed matter physics and chemistry by providing insights and predictions for electronic structure and properties of materials at the atomistic level<sup>1,2</sup>. These simulations enable researchers to predict a wide range of properties—such as electronic, magnetic, optical, mechanical, and thermal behavior—aiding in the design of novel materials for applications across diverse fields, including semiconductors, catalysts, energy storage systems, quantum materials, nanotechnology, and more<sup>3</sup>.

A multitude of powerful and widely used simulation codes have been developed in the past decades<sup>4</sup> and cross-verified<sup>5,6</sup>, to perform these calculations precisely and efficiently. Despite these advances, performing DFT

calculations remains a complex, multi-stage process that demands significant time and effort, particularly for non-specialists. By non-specialists, we refer to researchers whose primary expertise lies outside computational materials science, such as experimentalists, industrial R&D engineers, educators, and students. These users often face steep learning curves due to obstacles including: (1) software installation and configuration, (2) code-specific input file preparation, (3) parameter tuning, (4) results interpretation, (5) setup and utilization of high-performance computing (HPC) resources, and (6) validation and benchmarking against experimental data or established theoretical values. Furthermore, many property calculations require orchestrating long sequences of interdependent simulations—

<sup>1</sup>PSI Center for Scientific Computing, Theory and Data, Villigen PSI, Switzerland. <sup>2</sup>National Centre for Computational Design and Discovery of Novel Materials (MARVEL), Villigen PSI, Switzerland. <sup>3</sup>nanotech@surfaces Laboratory, Empa-Swiss Federal Laboratories for Materials Science and Technology, Dübendorf, Switzerland. <sup>4</sup>U Bremen Excellence Chair, Bremen Centre for Computational Materials Science, and MAPEX Center for Materials and Processes, University of Bremen, Bremen, Germany. <sup>5</sup>Dipartimento di Scienze Fisiche, Informatiche, Matematiche (FIM), Università di Modena e Reggio Emilia, Modena, Italy. <sup>6</sup>Nanoscience Institute, National Research Council (CNR-NANO), Modena, Italy. <sup>7</sup>Department of Physics and Earth Sciences, University of Parma, Parma, Italy. <sup>8</sup>Theory and Simulation of Materials (THEOS), École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland. <sup>9</sup>Center for Computational Chemistry, School of Chemistry, University of Bristol, Bristol, UK. <sup>10</sup>These authors contributed equally: Xing Wang, Edan Bainglass, Miki Bonacci, Andres Ortega-Guerrero.

✉e-mail: [xing.wang@psi.ch](mailto:xing.wang@psi.ch); [carlo.pignedoli@empa.ch](mailto:carlo.pignedoli@empa.ch); [giovanni.pizzi@psi.ch](mailto:giovanni.pizzi@psi.ch)

sometimes numbering in the tens or even hundreds. For instance, simulating phonon properties requires performing total energy calculations for each unique finite displacement (in supercells) as well as, if long-range dipole corrections are needed, for each applied finite electric field (in the unit cell). Even seasoned computational materials scientists (whom we will refer to as specialists in the following) routinely encounter such time-consuming, error-prone workflows that distract them from scientific insight.

Consequently, there is a growing demand for platforms that simultaneously (1) lower the entry barrier for non-specialists, enabling them to perform reliable simulations with minimal friction (for example, making advanced workflows accessible to experimentalists to help them interpret experimental results or guide the design of new experiments); and (2) provide specialists with robust, reproducible, and extensible infrastructures and advanced interactive analysis to enhance productivity in their simulations, as well as to allow them to incorporate their domain expertise into the platform, making it accessible to non-specialists. Achieving this goal requires a combination of: (1) ensuring that simulation codes use optimal computational settings with minimal user intervention; (2) implementing workflows that can detect and recover from common failures (e.g., switching to more robust algorithms in cases of convergence issues); (3) providing user-friendly user interfaces (UIs) to assist non-specialists in preparing inputs, submitting and monitoring workflows, and analyzing results; and (4) enabling easy sharing and reusing of data to foster transparency and collaboration.

Over the past decade, formal workflows have become increasingly important in computational materials science, enabling the automation of simulations and supporting large-scale, high-throughput studies that generate extensive databases of material properties<sup>7–15</sup>. Available workflow frameworks include AFLOW<sup>16</sup>, AiiDA<sup>17–19</sup>, ASR+ASE+myqueue<sup>20,21</sup>, Atomate+FireWorks<sup>22,23</sup>, Covalent<sup>24</sup>, Jobflow<sup>25</sup>, HTTK<sup>26</sup>, MISPR<sup>27</sup>, Pylon<sup>28</sup>, and QMPY<sup>29</sup>, among others. These frameworks facilitate researchers in managing automated sequences of simulations and often promote the FAIR (Findable, Accessible, Interoperable, and Reusable) data principles<sup>30</sup>, making it easier to share and reuse data generated by workflows among researchers, thus advancing the collective understanding of material behavior. One noteworthy advancement in this field has been the development of common workflows for computing material properties across different quantum engines<sup>20,31–33</sup>. These workflows provide a standardized approach to property calculations, improving reproducibility and facilitating direct comparison and verification between different DFT implementations<sup>6</sup>. Despite these advancements, existing platforms often require users to formulate workflows through scripting, necessitating programming skills that many non-specialists may not master. Additionally, these frameworks do not address the practical challenges, such as software installation and computational resource setup. Therefore, despite the advantages of workflows, significant barriers remain for non-specialists wishing to leverage these tools.

Graphical user interfaces (GUIs) could remedy this situation. However, developing GUIs for workflows traditionally requires in-depth knowledge of both workflow engines and graphical programming languages. Mature commercial offerings, such as AMS<sup>34</sup>, ASAP<sup>35</sup>, Mat3ra<sup>36</sup>, Materials Studio<sup>37</sup>, Materys<sup>38</sup>, MedeA<sup>39</sup>, QuantumATK<sup>40</sup>, Schrödinger<sup>41</sup>, and SimStack<sup>42</sup> have successfully tackled many of the practical challenges we outline above. Nevertheless, these platforms are often proprietary, and their closed-source nature could limit the ability of users to inspect, modify, and integrate the underlying code into the open-source ecosystem, thereby constraining transparency, reproducibility, and full compliance with FAIR principles. On the open-source front, Quantum ESPRESSO (QE) provides its own GUI, PWgui<sup>43</sup>, and several third-party tools have also been developed, such as BURAI<sup>44</sup>, which are excellent for generating input files for single calculations. Other codes, like GPAW<sup>45</sup>, also provide dedicated GUIs. While invaluable, these tools are typically limited to specific codes and often focus on setting up individual calculations rather than managing complex, multi-stage workflows. Other frameworks—such as CINEMAS<sup>46</sup> and ALKEMIE<sup>47</sup>—have emerged, providing user-friendly GUIs and integrated

environments for advanced materials simulations. However, these solutions generally require a local installation, highlighting the ongoing need for more flexible, web-based platforms that can accommodate both local and cloud-based usage while remaining widely accessible to the research community.

In our previous work, we developed AiiDALab<sup>48</sup>, a web-based platform that enables computational scientists to package scientific workflows and computational environments into user-friendly applications (apps). AiiDALab integrates AiiDA workflows with graphical interfaces (based on Jupyter notebooks) for execution and data analysis, making them accessible to a broader audience. In this work, leveraging the AiiDALab platform, we identify a common structure when designing an app and GUI for DFT calculations, and develop a generic system that can serve as a template for other simulation codes. We demonstrate this concept by implementing a feature-rich app specifically designed for QE<sup>49–51</sup>, an open-source and widely used DFT simulation package<sup>4</sup>. In the following sections, we describe the architecture and general structure of the app, presenting some implementation details including the integration with AiiDA and AiiDALab. We then demonstrate how it addresses the challenges faced by non-specialist users in performing DFT simulations by showcasing the app's features through example use cases. These features leverage several AiiDA workflows recently developed for computing relevant materials properties (ranging from electronic structure calculations to vibrational and spectroscopic properties) and for supporting and automating advanced computational methods (such as DFT+*U*+*V* and Wannier functions).

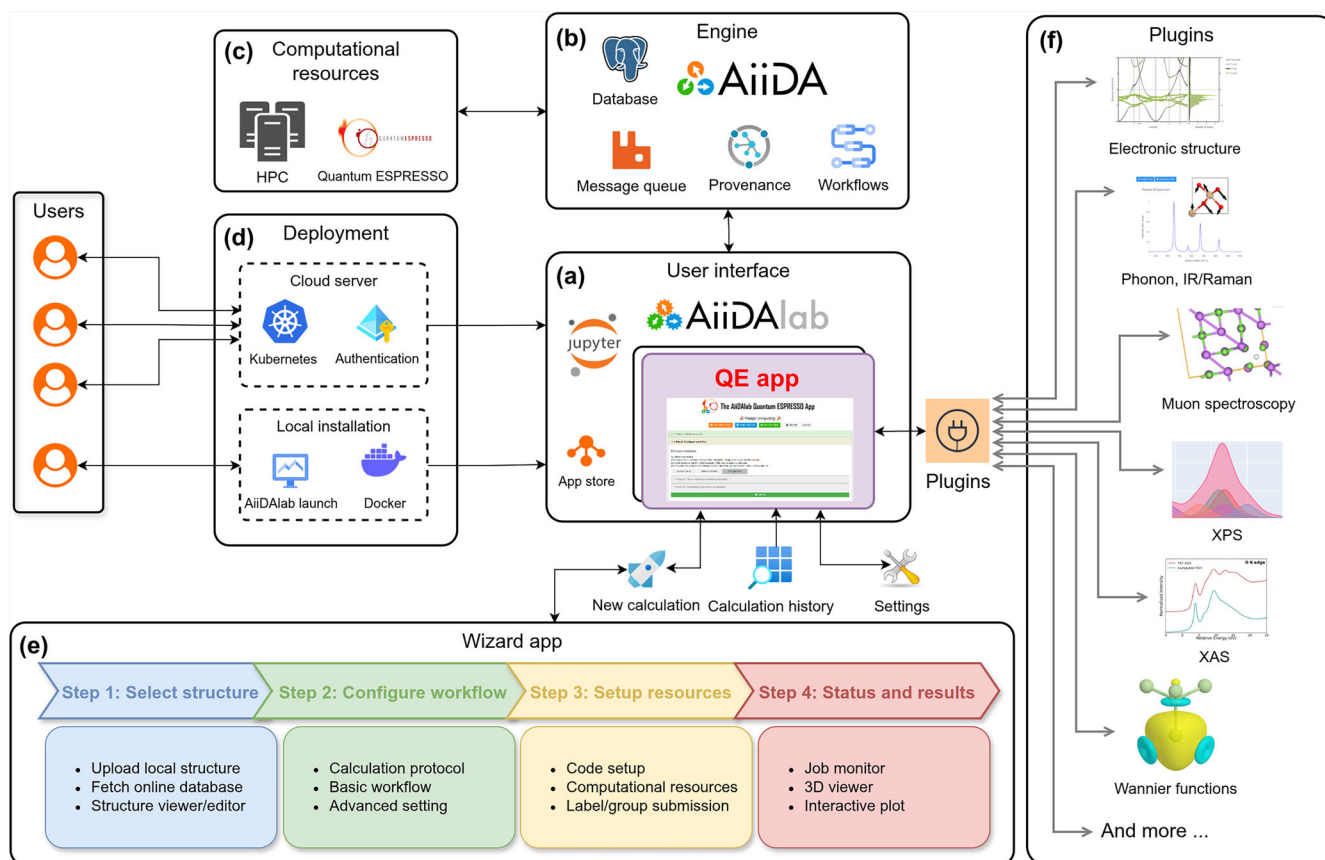
## Results

### Architecture overview

The QE app architecture aims to provide a turn-key solution for materials scientists to run DFT calculations seamlessly, addressing key challenges outlined in the introduction. As illustrated in Fig. 1, the app consists of several key components to handle different aspects of the simulation workflow: (a) a user interface (UI) built within the AiiDALab ecosystem using Jupyter notebooks and the appmode plugin<sup>52</sup>, (b) an AiiDA-based backend engine responsible for workflow management and data provenance that interacts with (c) external HPC resources, (d) flexible deployment options, including both cloud-based and local installations, (e) a “wizard” app guiding users through the simulation pipeline, and (f) a plugin system for extensibility, providing access to several advanced workflows to compute different material properties, combined with specialized post-processing tools.

One of the key challenges in designing the QE app is that different material properties require distinct inputs, workflows, and outputs. A straightforward but suboptimal approach would be to create a dedicated UI or sub-app for each property, but this leads to a fragmented user experience and increased maintenance complexity. In contrast, this app adopts a unified design strategy that enables users to perform calculations of different material properties using QE within a single, coherent interface. At the core of the app's UI design is the Input-Process-Output (IPO) model<sup>53</sup>, which is widely used in workflow management systems, where it provides a clear and organized approach to handling tasks, ensuring systematic processing across different stages<sup>54,55</sup>. The app implements the IPO model at two interconnected levels. On the backend, a top-level AiiDA WorkChain orchestrates property-specific sub-workflows, managing inputs, processes, and outputs in a standardized way. A detailed schematic of this IPO workflow architecture is provided in the **Input-Process-Output model** section of the Supplementary Information. On the frontend, this logic is mirrored by a wizard UI that guides users through the simulation process in a structured and intuitive way (see Fig. 1e). The wizard ensures that all necessary inputs are provided before proceeding to the next step, but also allows users to revisit and modify previous steps if needed, offering both flexibility and control during the setup process. The details of the wizard UI are provided in a later section.

In academic development projects, rapid prototyping often requires incorporating new algorithms and protocols into existing workflows. The QE app addresses this need with a plugin-based architecture that allows



**Fig. 1 | Overview of the QE app's architecture.** **a** The UI is built within the AiiDALab environment using Jupyter notebooks and the `appmode` plugin, providing an intuitive front end for simulation setup and results analysis. **b** The backend engine is powered by AiiDA, which handles workflow orchestration, data management, and provenance tracking. **c** The app connects to external high-performance computing (HPC) resources for running simulations. **d** The app can be deployed on cloud-based deployments using Kubernetes and authentication layers, or locally using the

AiiDALab launch tool or Docker. **e** A modular wizard UI guides users through the four main steps of the simulation process. **f** The plugin system enables extensibility by allowing users to activate specific property calculations or post-processing tools, such as electronic structure, vibrational properties (IR/Raman), muon spectroscopy and resting sites, XPS, XAS, Wannier functions and more. The full list of available plugins is discussed later in the text.

flexible addition of new functionalities. Material property calculations, such as band structures or projected density of states (PDOS), are implemented as plugins, each operating as a self-contained unit within the IPO model, managing its own input, process, and output. The main app provides a unified API and mounts the components of the plugins, ensuring seamless integration. This design preserves a consistent interface across all functionalities, allowing the main application to focus on core features shared by all plugins. It also simplifies the work of plugin developers, who can build and test their components independently, without needing to modify the core application. Plugins can be developed and maintained either within the main code base or as standalone packages. By extending functionality through plugins without altering the core structure, the QE app maintains a single, streamlined interface for setup, execution, and analysis, reducing the learning curve and delivering a smooth, consistent user experience.

### App accessibility

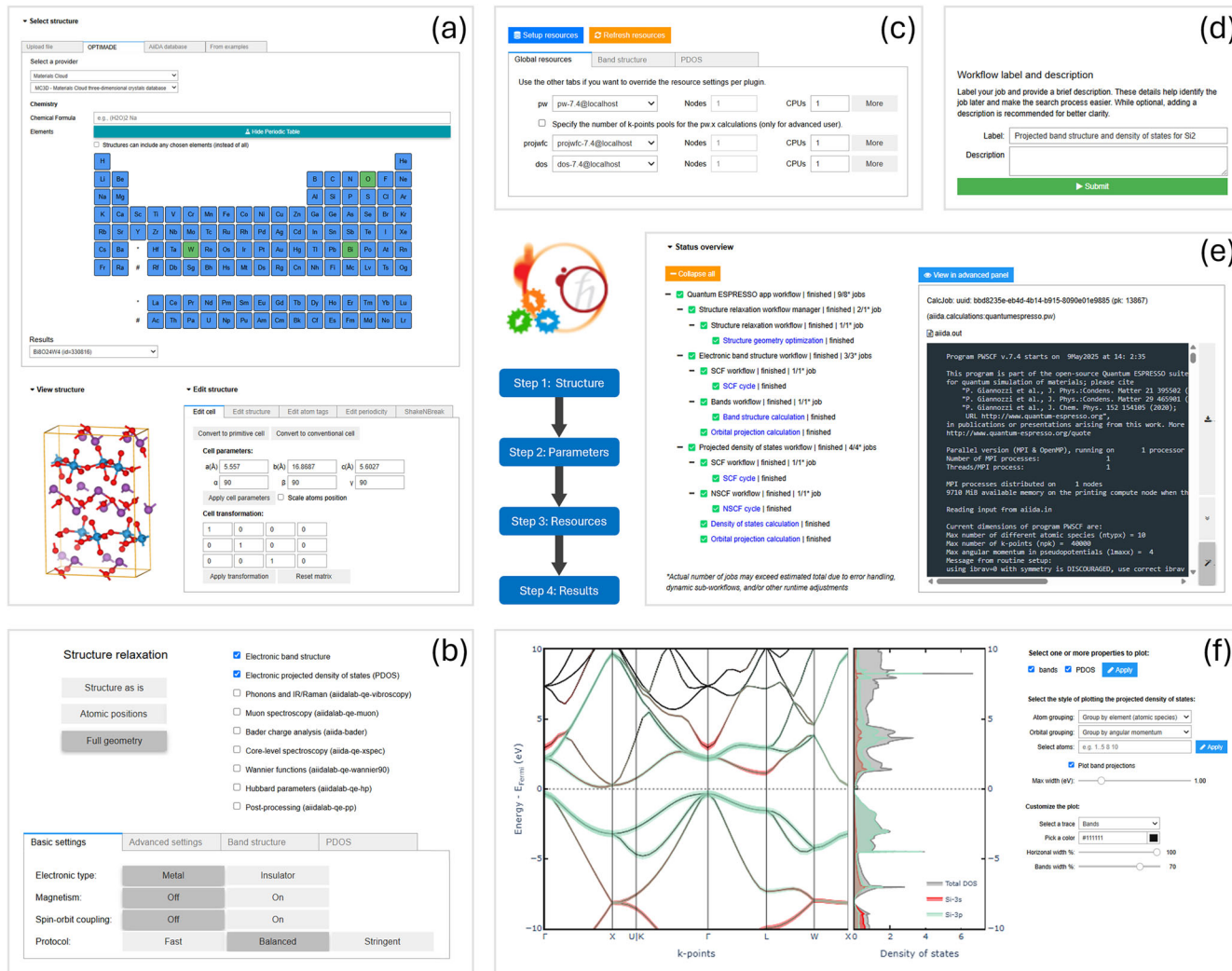
The QE app can be used both locally and via the cloud, offering flexibility for different user needs and environments. For local workstations, the app can be installed via the `aiddalab-launch` utility (see the **Code availability** section), which automates the process of retrieving the appropriate Docker image and launching a containerized instance of AiiDALab with the QE app pre-configured. This option enables users to run the application in a fully isolated and reproducible environment on their own machines.

For cloud access, AiiDALab's centralized Software-as-a-Service (SaaS) model allows users to access the platform via a web browser without the need for local software installations. This model supports scalable

deployment, from individual users and small research groups to larger institutional infrastructures. We provide a publicly accessible demonstration instance of the QE app at (<https://demo.aiddalab.io>). The server is deployed via a dedicated Kubernetes infrastructure on the Microsoft Azure service and comes with the app and its core plugins pre-installed. Users can readily explore the app's functionality, inspect pre-computed examples, and submit test calculations. This public demo is currently maintained under the NCCR MARVEL. The goal of this demo is to facilitate first-time users in getting a first impression of the functionality of the app and of the platform without the need of any local installation. Should the public server become unavailable in the future, the app's continued accessibility is ensured through the local installation option via `aiddalab-launch` described earlier, which provides a permanent way for users to run the tool on their own machines.

### Leveraging AiiDA for workflow management

Workflows help standardize simulations, minimize manual intervention, and improve reproducibility, making them an essential component of the app's turn-key solutions. To manage its calculation workflows, the QE app leverages the AiiDA workflow management system, which serves as the backend engine for job execution, data provenance, and automated workflow coordination. AiiDA manages communications with HPC machines, transferring files, interacting with the job scheduler, and submitting and monitoring calculations. Besides, AiiDA supports a broader ecosystem of community-maintained plugins, available through an official registry (<https://aiddatteam.github.io/aidda-registry/>), which covers a wide range of



**Fig. 2 | The QE app’s wizard UI steps. a** structure selection step, where users can select, view, and edit the input structure of the calculation; **b** workflow configuration step, where users specify the main workflow steps (structure relaxation and properties to compute) and customize the parameters of each of them; **c, d** resource

selection and workflow submission step, where users select codes and computational resources, provide optional metadata, and submit the workflow; and **e, f** monitoring and analysis step, where users can monitor the running calculations, plot results from completed calculations, and customize plots interactively.

simulation codes, including both calculation interfaces and automated workflows.

The app uses the `aiida-quantumespresso` plugin package (see the **Code availability** section), which provides a collection of pre-configured AiiDA calculation plugins to prepare the input and parse the output of several of the QE executables. These calculation classes serve as building blocks for more complex workflows that link multiple computations together, enabling users to perform sophisticated simulation tasks with minimal manual input.

AiiDA also supports automatic restarts and error handling mechanisms. Building on this, the `aiida-quantumespresso` plugin implements specific recovery strategies tailored to QE calculations. For example, if a calculation fails to reach electronic convergence, the workflow can automatically adjust key parameters, such as reducing the mixing ratio for self-consistent iterations. Similarly, if a job is interrupted due to exceeding the allocated walltime, it can be resumed cleanly using the last output structure and charge density. These recovery strategies are embedded in the workflow logic and require no user intervention, reducing manual troubleshooting and increasing the reliability of complex simulations. In addition, AiiDA implements a caching mechanism that allows the reuse of results from previous calculations submitted with identical inputs, thus avoiding redundant computations and promoting consistency across multiple runs.

To streamline the process of preparing simulations, the app exposes other utilities from `aiida-quantumespresso` to facilitate simulation setup and customization, for example, to automate **k**-point mesh generation tailored to the size of the selected system.

### QE app wizard UI

The app’s wizard UI serves as a guide for users as they construct and submit a workflow. Throughout the process, the wizard ensures that all necessary information is provided before progressing. Each step in the wizard consists of multiple panels that allow users to refine their input as needed. Prior to submission, users can also return to a previously confirmed step to modify any input data. The UI divides the entire process into four main steps.

The first step is selecting a structure. This step includes tools to select, view, and modify the input structure (Fig. 2a). Users can upload their own structure (in any standard format, such as XYZ, CIF, etc.), or choose a structure from online databases compliant with the OPTIMADE standard<sup>56,57</sup> (such as the Materials Project<sup>8</sup>, the Materials Cloud<sup>58</sup> and the Alexandria Materials Database<sup>59,60</sup>), from their own local AiiDA database, or from a set of example structures. The structure viewer allows for inspection, providing an overview of structure properties (cell parameters, volume, space/point group) and with a set of tools to interact with and/or manipulate the atomistic model: edit the structure cell (primitive, conventional, and

supercell transformations), periodicity, (surfaces, molecule definition), and atoms (removal, tagging). Other editing features include structural modifications, such as introducing point defects in 2D materials and applying bond distortions or random displacements to explore metastable configurations of point defects, following the approach implemented in the ShakeNBreak package<sup>61</sup>.

The second step is configuring the workflow. With an input structure selected, users can now configure the workflow, optionally including a structure geometry optimization step, selecting from a host of properties of interest to compute (see **Plugins** section, also in Supplementary Information), and customizing calculation parameters (Fig. 2b). To simplify this process, the app splits the parameter customization step into basic, advanced, and property-specific panels. The basic panel streamlines the process by abstracting much of the complexity via a set of top-level controls, including a protocol selector to set most input parameters balancing calculation speed and accuracy<sup>62</sup>. Full flexibility in customizing calculation inputs is retained via the advanced panel. In this panel, the user can choose among various pseudopotential families, including the Standard Solid-State Pseudopotentials (SSSP; efficiency or precision)<sup>63</sup> and PseudoDojo (standard or stringent)<sup>64</sup>, or alternatively upload custom pseudopotential files directly via the GUI. Further details of each panel are provided in the Workflow parameter settings panels section of the Supplementary Information.

The third step is choosing computational resources and submitting the workflow. Users proceed to choose the computational resources to use by AiiDA to run the calculation (Fig. 2c). The app provides a straightforward UI for selecting AiiDA Code instances—references to executables on local or remote machines—and specifying the computational resources to be allocated per code (e.g., nodes, CPUs). Additional codes may be set up in a separate notebook (see the Computational resources setup section of the Supplementary Information). Once the resources have been chosen, users may customize the auto-generated workflow label and optionally provide a description to attach to the workflow (Fig. 2d). When ready, users can submit the workflow. AiiDA will handle submission in the background.

The fourth step is monitoring and viewing workflow results. After submitting a workflow, the app redirects users to the status panel of the results step (Fig. 2e), where they can monitor the calculations in real-time. Users can switch to the summary panel to inspect a summary of the selected inputs. As calculations complete, results from each calculation submitted by the app's workflow are made available for analysis in dedicated sub-panels of the results panel (Fig. 2f). Lastly, when the workflow is finished, users can download from the summary panel either a zip file including the raw input and output data of all workflow steps, or directly a single AiiDA archive (including the full provenance graph) for further inspection or sharing. The latter option (AiiDA archive) allows users to import the workflow into another AiiDALab instance and continue inspecting the results via the QE app GUI.

The app's plugin-based architecture provides several entry points to extend the core UI described above. Plugin developers can introduce additional UI components, including structure importers and editors, property-specific parameter settings panels, computational resource panels to override global resources per property-specific code, and property-specific results panels. The following section provides examples of such plugin components.

### Plugins: overview

The QE app's plugin-based architecture provides a flexible framework for extending the platform with specialized functionalities, ranging from electronic-structure analysis to advanced spectroscopic techniques. Each plugin incorporates its own workflows and UI components, yet remains fully integrated within the app's unified IPO model and the broader AiiDALab environment. This modular design not only streamlines maintenance and testing but also encourages community-driven extensions. Below, we highlight the key aspects of two plugins, one provided as a core plugin included in the app's code base (electronic structure) and one developed as an external plugin (vibrational spectroscopies). We also

provide a short summary for each of the currently supported app plugins. The plugin packages are listed in the **Code availability** section.

### Plugins for the electronic structure: band structures and (projected) densities of states

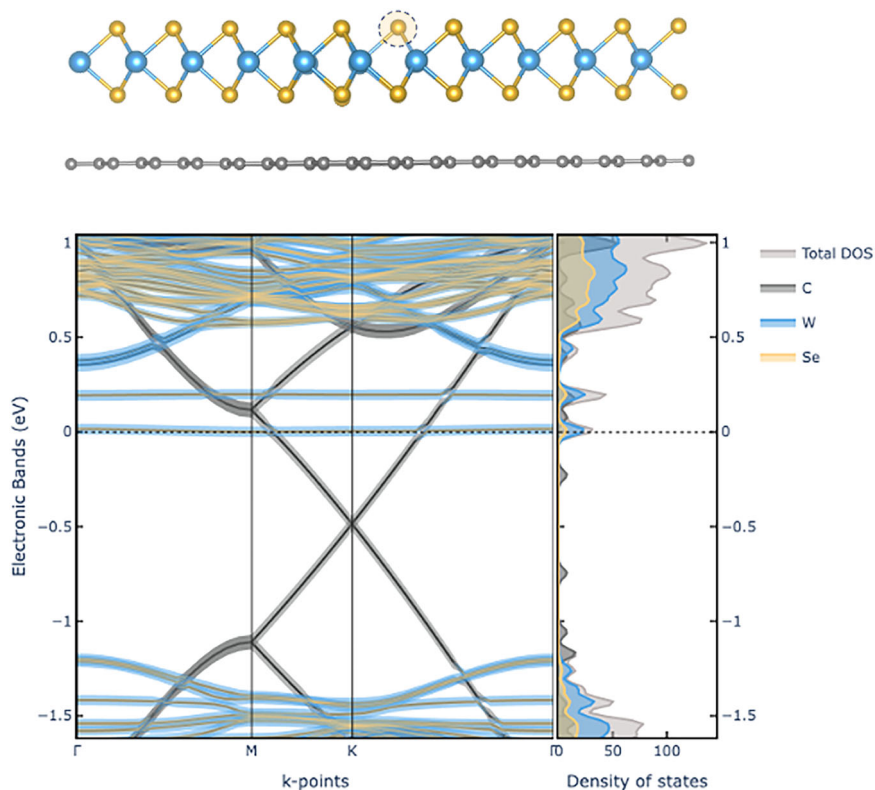
Electronic band structure theory is fundamental for interpreting material's behavior, including electrical transport and optical properties, making it a cornerstone of solid-state physics<sup>65</sup>. In the QE app, band-structure calculations are streamlined through a dedicated plugin that guides the users from processing to final visualization. The process begins with supplying an initial crystal structure, which can be either a representation of an experimentally known structure or a computationally optimized one. The app automatically defines the band-structure path based on the structure. For 3D materials, the app leverages existing tools—such as seekpath<sup>66</sup>—to standardize the lattice and identify a *k*-point path based on well-known symmetry lines<sup>67</sup>. In low-dimensional cases (e.g., 2D monolayers), the app detects the system symmetry and selects the appropriate reciprocal paths that capture the most relevant dispersion features.

Beyond computing the band energies, the plugin incorporates projected band structure analysis, also known as “fat bands”, which quantifies the contribution of specific atomic orbitals to each energy level; an example is shown in Fig. 3f. These projections help reveal crucial effects, such as orbital hybridization, localized defect states, or spin-splitting phenomena<sup>68</sup>. In addition to fat bands, where projection onto atomic orbitals is studied only along the high-symmetry paths in reciprocal space where the band structure is computed, PDOS calculations provide an additional perspective on the electronic structure of materials by revealing how specific atoms or orbitals contribute to the total density of states (DOS), e.g. helping in identifying which atomic orbitals dominate states near the Fermi level, thereby offering insights into the origins of electronic, magnetic, or optical properties. In the QE app, the PDOS workflow integrates four core steps: self-consistent field (SCF), non-self-consistent field (NSCF), total DOS, and PDOS, into a streamlined sequence that automatically generates necessary wavefunction files and appropriately handles large intermediate data. In particular, orbital-projected DOS calculations decompose the total DOS into site- and orbital-specific contributions, highlighting the role of, for instance, *d*-orbitals in transition-metal oxides or *p*-orbitals in organic semiconductors. Moreover, both the band structure and PDOS plugins support spin-orbit coupling (SOC) calculations, enabling an accurate description of relativistic effects in materials where SOC plays a significant role, such as heavy-element semiconductors and topological insulators, and allowing users to analyze SOC-induced modifications in the DOS and PDOS. Within the standard app UI, users can also configure advanced parameters, such as *k*-point densities, smearing schemes, or energy ranges. This flexibility is particularly important for materials where fine spectral features can strongly influence theoretical predictions of conductivity or band alignments. The final PDOS data are presented via interactive plots, allowing for immediate visual inspection of peaks and energy offsets relative to the Fermi level. The PDOS can also be displayed in tandem with the band structure, further clarifying which states govern conduction or valence behavior, as we illustrate in Fig. 3. The interactive panel of the plugin enables users to adjust visual elements, isolate specific states, and export publication-quality figures or the underlying numerical data. By automating complex workflows in a reproducible environment, this plugin significantly reduces the computational barrier for advanced electronic structure studies of complex materials.

### Plugins for vibrational spectroscopies: phonons and Raman/Infrared spectra

Vibrational spectroscopies are among the most powerful methods for materials structure investigation, via fingerprint atomic vibrations probed by using several types of radiation, such as light. The simulation of vibrational properties in the QE app is available through the `aaidalab-qe-vibrospectroscopy` plugin. The app has been successfully applied to investigate the vibrational and optoelectronic properties of BaZrS<sub>3</sub>, highlighting its

**Fig. 3 | Schematic illustration of a selenium (Se) vacancy in a WSe<sub>2</sub>/graphene heterojunction (top) and the corresponding projected band structure and projected density of states (PDOS) (bottom), computed using the QE app with spin-polarized DFT using the PBE exchange-correlation functional<sup>116</sup>, and including SOC. This study explores the hybridization of Se vacancy states with graphene and their interactions. The in-gap defect states, which are non-dispersive and exhibit a gap due to spin-orbit coupling (SOC) effects, are located between the conduction and valence bands of WSe<sub>2</sub>, as discussed in ref. 117.**



practical utility in ongoing research and materials characterization efforts<sup>69</sup>. This is accomplished via the specific workflow, called *VibroWorkChain*, which orchestrates the submission of AiiDA workflows provided by the *aiida-vibrospectroscopy*<sup>70</sup> and *aiida-phonopy*<sup>70-73</sup> plugins. Specifically, phonon properties are obtained using a finite-difference approach<sup>67</sup> through multiple SCF calculations, involving finite atomic displacements from equilibrium positions in supercells. Large enough supercells are needed to accurately interpolate phonon dispersions, equivalent to using dense  $\mathbf{q}$ -point meshes in density-functional perturbation theory (DFPT)<sup>74</sup>. If long-range non-analytic corrections (NACs) are required, the workflow will also compute dielectric properties via finite applied electric fields<sup>75</sup> simulations. This approach differs from the one implemented in the *Phonon* code of the QE package<sup>74</sup>, and enables calculations with any functional, such as hybrid functionals or DFT+ $U$ (+ $V$ )<sup>70</sup>. Various types of simulations can be performed based on the desired outcome: (1) full characterization of phonon properties, including phonon bands, phonon PDOS, thermodynamics, and infrared (IR) and Raman spectra; (2) phonon properties for non-polar materials (i.e., NACs are not computed); (3) IR/Raman spectra from  $\Gamma$ -point normal modes of the unit cell; (4) dielectric properties (dielectric and Raman tensors, Born effective charges). As shown in Supplementary Fig. S4, options are provided to users by means of a drop-down menu, together with options to select (or automatically determine) the supercell size needed for finite atomic displacements and the symmetry tolerance used to detect unique atomic displacements (thus reducing the cost of the simulation). As a rule of thumb, we suggest a supercell with lattice parameters of at least 15 Å, which is typically enough to capture all the relevant atomic displacements to be considered in the phonon simulation, i.e., for force-constant matrix elements to decay to zero for atom pairs beyond this distance. An estimate of the number of finite displacement supercells can be performed, providing an indication of the number of simulations that will be submitted, and thus of the total computational cost.

If phonons are computed (i.e., if users select one of the first two calculation options), the inelastic neutron scattering (INS) dynamical structure factor can additionally be computed as a live post-processing functionality

in the output panel. Specifically, INS observables for both single-crystal and powder samples are calculated by means of the *Euphonic* package<sup>76</sup> in a format directly comparable with experimental spectra. Users also have the possibility to provide a specific reciprocal space path or plane to facilitate comparison of the simulation results with a specific experimental setup and measurements.

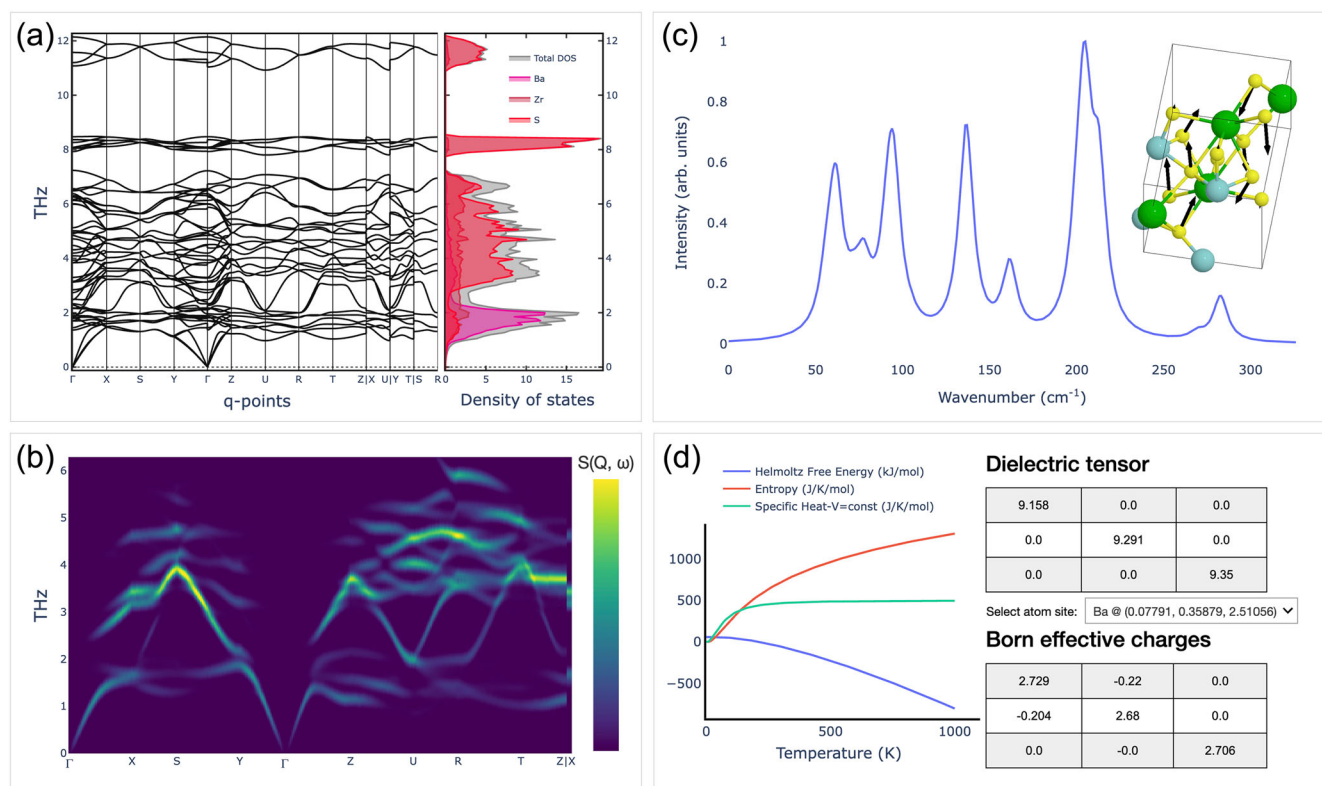
Selected screenshots of the plugin interface are shown in Fig. 4 for the previously mentioned BaZrS<sub>3</sub> system<sup>69</sup>, showing that the results tab is subdivided into different sub-tabs, each of them dedicated to a specific output of the simulations: (i) phonon band structure, phonon PDOS and thermodynamic quantities, see panel (a); (ii) INS results, to be computed live, see panel (b); (iii) IR and Raman active modes, together with their interactive 3D animations, see panel (c); (iv) thermodynamic and dielectric properties, see panel (d). All the data shown in the panels can be downloaded and analyzed with additional external tools, if needed.

### Additional plugins

Beyond the two headline examples, the app supports a growing suite of specialized plugins designed to automate and visualize advanced materials simulations. Below, we summarize the key functionalities and materials properties addressed by each plugin. Representative result panels from these plugins are shown in Fig. 5, while setup options and further technical details are provided in the **Plugins** section of the Supplementary Information.

The XAS plugin facilitates the computation of X-ray absorption near-edge structure (XANES), a technique sensitive to the local chemical environment of absorbing atoms<sup>77</sup>. The plugin is based on a reciprocal-space pseudopotential scheme, as implemented in the *xspectra.x* code<sup>78,79</sup> of QE, and accounts for different core-hole treatments, symmetry-analysis for inequivalent site selection, and lifetime broadening<sup>79</sup>. Users can select the absorbing element and inspect both total and site-resolved XANES spectra. Interactive controls allow adjusting broadening parameters and exporting spectra for further analysis.

The XPS plugin enables the calculation of core-level binding energies using a total-energy difference approach<sup>80-82</sup>, widely used to interpret



**Fig. 4 | Results panels for the aiidalab-qe-vibrosopy plugin. a** Phonon dispersion plot for BaZrS<sub>3</sub>, together with the phonon DOS (aligned in energy with respect to the phonon bands). **b** Computed INS spectrum. **c** Single crystal Raman spectrum and

modes animation. **d** Thermodynamic properties, dielectric tensor and Born effective charges.

experimental XPS spectra and identify chemical shifts across atomic sites. Users can select target atoms, specify core-hole pseudopotentials, and set spectral broadening parameters. The results panel displays both the computed binding energies and their site-resolved chemical shifts in an interactive table and spectrum viewer, allowing comparison with experimental reference data.

The muon spectroscopy plugin targets the prediction of muon stopping sites and their interaction with the magnetic environment, crucial for interpreting muon spin relaxation ( $\mu$ SR) experiments<sup>83</sup>. By simulating a dilute positive muon in a material using DFT+ $\mu$ <sup>84–86</sup>, users can identify likely stopping sites and compute associated local fields. The setup panel lets users configure the supercell size, charge state, and number of trial sites (inputs for the aiida-muon plugin<sup>87,88</sup>). The results include a ranked list of candidate sites with respect to relative total energy, 3D site visualization and, optionally, the muon spin polarization functions computed for different crystal orientations via the UNDI package<sup>89</sup>.

The Wannierization plugin automates the generation of maximally localized Wannier functions (MLWFs), used to analyze bonding characters, build tight-binding models, and study topological invariants<sup>90–92</sup>. Users can choose between automated Wannierization schemes [e.g., the selected columns of the density matrix (SCDM) method<sup>93,94</sup> or the projectability disentanglement Wannier functions (PDWF) method<sup>95</sup>] and adjust the energy windows<sup>95</sup>. It is also possible to compute the Fermi surface using Wannier interpolation, allowing for dense Brillouin zone sampling (k-point distance down to a few hundredths of reciprocal angstrom). For the generated Fermi surface, it is possible to calculate de Haas–van Alphen (dHvA) oscillation frequencies, facilitating direct comparison with experimental measurements<sup>96,97</sup>. The result panel compares DFT and Wannier-interpolated bands, reports total and component spreads, offers a 3D visualization of individual Wannier functions with atoms, and provides a plot with computed dHvA frequencies. Files containing the tight-binding

model in the MLWF basis set and the Fermi surface are available in the “Downloads” section of the result panel.

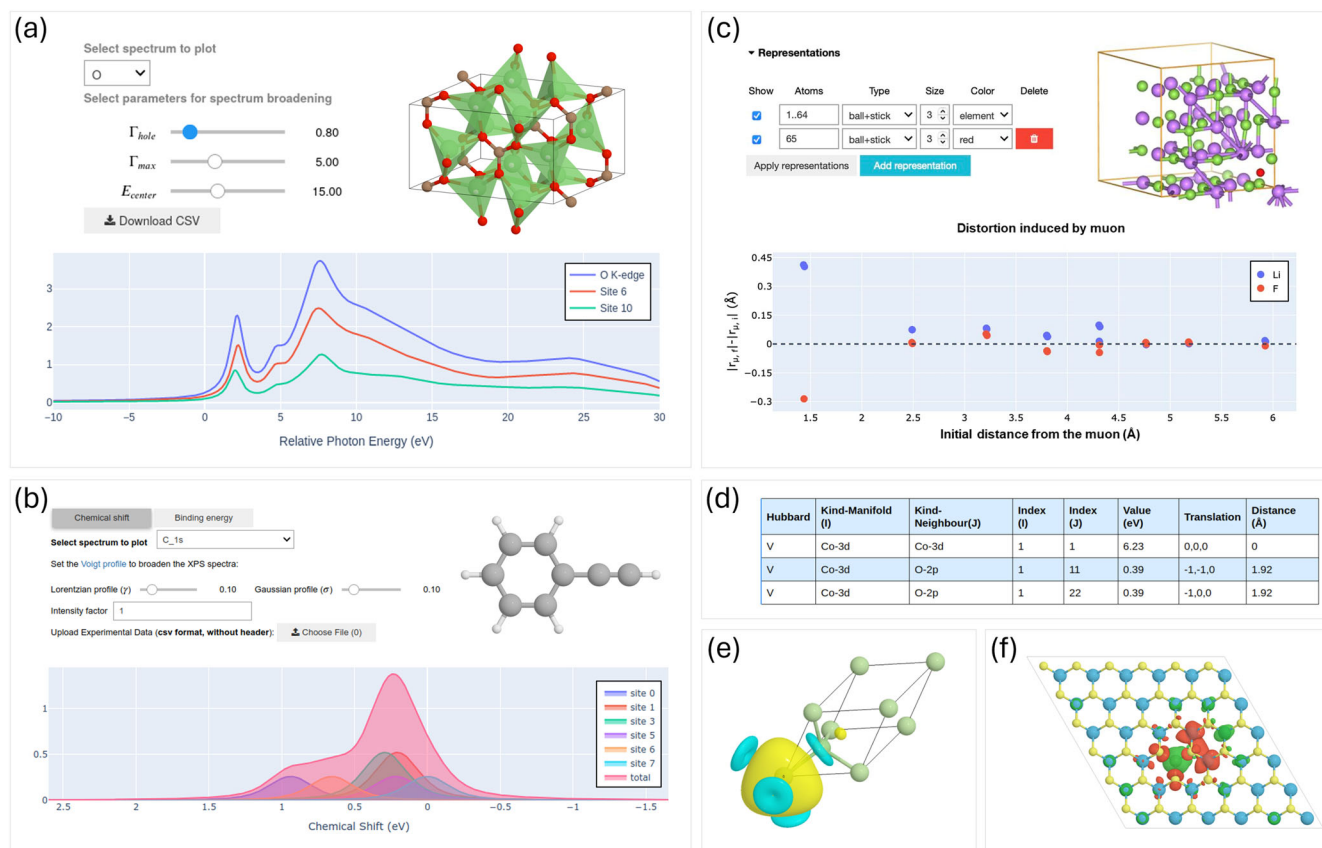
Accurate determination of on-site  $U$  and inter-site  $V$  Hubbard corrections is essential for treating transition-metal and rare-earth compounds containing partially filled and localized  $d$  and/or  $f$  electrons<sup>98–101</sup>. The Hubbard parameters plugin interfaces with the aiida-hubbard workflows<sup>102</sup>, which efficiently orchestrate DFPT-based calculations<sup>103,104</sup> to compute these parameters from first principles using the hp.x code of QE<sup>105</sup>. Users can toggle between “one-shot” or “self-consistent” modes, choose atom pairs for inter-site corrections, and specify convergence thresholds. Results include a summary table of the computed Hubbard parameters and an interactive structure viewer linking each value to its corresponding site or atom pair.

The post-processing plugin enables real-space visualization of properties, such as charge and spin densities, wavefunctions, electrostatic potential, integrated local density of states (ILDOS), local density of states at specific energies, and STM images using the pp.x and Critic2 codes<sup>106</sup>. Users can select previously completed calculations, choose the target quantity, and define isosurface or plane-plot parameters. Results are displayed using an integrated 3D viewer powered by weas-widget<sup>107</sup>, providing an intuitive way to correlate electronic structure features with spatial distributions.

The Bader charge analysis plugin, by partitioning the charge density into atomic basins (based on the Bader code<sup>108</sup>), provides chemically meaningful net atomic charges—useful for understanding charge transfer, oxidation states, or catalysis. The plugin processes charge density data from completed calculations and presents the atomic charges in a sortable, interactive table alongside the structural model.

#### App utilities

Apart from the core and plugin functionalities encapsulated in the wizard UI, the QE app also provides a set of utility pages for the following



**Fig. 5 | Overview of result panels from various plugins. a** XAS plugin showing the O K-edge XANES spectrum of crystalline Li<sub>2</sub>CO<sub>3</sub>, including site-resolved and total contributions. **b** XPS plugin applied to phenylacetylene (C<sub>8</sub>H<sub>6</sub>) in gas phase, illustrating the C 1s core level shifts and their relation to atomic environments. **c** Muon spectroscopy plugin showing a candidate muon site in LiF and the distortions induced by the muon to the hosting lattice. **d** Hubbard  $U + V$  plugin calculating on-

site and inter-site interactions in LiCoO<sub>2</sub>, including a table of computed parameters and their atomic associations. **e** The Wannier function plugin showing isosurfaces of one of the maximally localized Wannier functions in gallium arsenide. **f** Post-processing plugin visualizing spin density data in a sulfur vacancy in a monolayer of MoS<sub>2</sub>.

operations. This includes browsing calculation history, where users can browse, filter, and manage all previously submitted jobs (see Supplementary Fig. S15); installing app plugins, which allow users to find, install, and manage plugins, customizing their computational environment as needed (see Supplementary Fig. S16); configuring computational resources, where users can define local and remote machines and codes, as well as browse (and optionally disable) existing codes (see Supplementary Fig. S17); downloading example calculations, enabling users to import example AiiDA archives into their database with one click. We provide a host of examples spanning core and plugin functionality, covering most common use cases (see Supplementary Fig. S18). Further details on each utility are provided in the **Utilities** section of the Supplementary Information.

## Discussion

The QE app and its underlying AiiDA ecosystem demonstrate in practice how the FAIR principles can be extended beyond data only, to include the full life-cycle of a first-principles simulation, including the workflows that generate the data, the codes that execute them, and the interactive tools used to analyze the results<sup>7</sup>. Below, we map each FAIR dimension onto concrete features that are already available in the QE app or planned for the near future, underscoring the added value for both end-users and developers.

In terms of findability, every calculation launched from the QE app is automatically stored in the AiiDA provenance graph. Users can export the complete graph—including inputs, outputs and metadata—as a single AiiDA archive file. These archives can be deposited in the *Materials Cloud Archive*<sup>58</sup>, which offers persistent DOIs and a rich web interface that lets

third parties browse the full data lineage and inspect input and output files. At the workflow layer, AiiDA's entry-point mechanism registers every workflow, parser and calculation plugin with a unique name and provides a plugin registry for plugin developers to register their workflows. At the plugin layer, the app provides a *Plugin management* page, where users can discover and install additional plugins from the community.

Accessibility is achieved through flexible deployment modes. The QE app is available in both cloud-based and local configurations, lowering the entry barrier for non-specialists. In the cloud, it can be accessed directly through a web browser without installation or setup. The cloud deployment is managed by Kubernetes and JupyterHub, which handle authentication and authorization via standard OAuth2 protocols, ensuring secure and scalable access. To provide an immediate hands-on experience, we also offer a publicly accessible demonstration instance at <https://demo.aidalab.io>. For local use, the app can be launched using the `aidalab-launch` utility, which sets up a pre-configured, containerized environment. These deployment options (see App accessibility section) ensure flexibility and broad accessibility across user environments.

Interoperability is a key strength of the app's architecture. While the app currently supports only QE, its architecture is designed to be interoperable by serving as a generic template that can be adapted to other DFT codes. The core components are code-agnostic: the AiiDA-powered backend, the AiiDALab-powered frontend, the IPO model, and the modular wizard UI. Adapting this framework for a new code, therefore, does not require re-engineering the entire application. Instead, a developer can focus on implementing code-specific input schema and graphical interface extensions, while the overall architecture remains unchanged. By

exploiting the common workflow interfaces for over ten DFT codes that have been defined and implemented in AiiDA<sup>6,31</sup>, developers can reuse the app's structure to build similar applications for other quantum engines. For example, we are currently exploring the adaptation of this framework to the CP2K code by integrating the independently developed `aiidalab-empa-surfaces` app<sup>109</sup> into the same structural architecture used in the app. The app's architecture is also serving as a model for a DFT-code-agnostic GUI developed as part of the PREMISE project of the ETH Board (<https://ord-premise.org/>). Furthermore, the interactive widgets and results analysis tools integrated into the app have proven invaluable not only for non-specialists but also for seasoned computational materials scientists. To maximize their utility, we plan to further modularize these tools, allowing users to employ them outside of the app's main interface, such as in a standard Jupyter notebook. These tools can also be utilized on other platforms, like the Materials Cloud website<sup>58</sup> and the OSSCAR platform<sup>110,111</sup>.

Reusability and reproducibility are ensured through open-source licensing and the integration of AiiDA's provenance tracking. All components of the QE app, including the main interface, the underlying AiiDA plugins, and the analysis widgets, are released under permissive open-source licenses and hosted on public GitHub repositories (see the **Code Availability** section), maximizing reusability and transparency. By using AiiDA for workflow management, the app ensures that all inputs, processes, and outputs are automatically stored in a structured, queryable database. When a calculation is restarted (e.g., to resolve SCF convergence issues), the updated settings are also recorded, preserving the full decision-making process. Even publication-quality figures generated via the interactive plotting widgets are fully reproducible with a single click, as the complete widget state (e.g., axis ranges, color maps, structure viewer settings) is stored alongside the corresponding AiiDA node. While reproducibility goes beyond the original FAIR acronym, our design makes it a practical reality and aligns with the provenance-focused spirit of the FAIR guidelines. Furthermore, the app's plugin architecture ensures extensibility and modularity, allowing users and developers to share and reuse specialized workflows for diverse material properties and advanced analytical techniques, thus facilitating broader community collaboration.

Apart from FAIR, the development of the QE app also underscores the importance of user-friendly interfaces. These interfaces lower entry barriers for non-specialists and accelerate the onboarding process. For example, the in-app guide system, which overlays context-specific information and "click-here-next" instructions on top of any panel, trains the user in place, avoiding the need for external manuals (see Supplementary Fig. S14). Ongoing development efforts aimed at decoupling the UI from the AiiDA backend via modern UI frameworks and dedicated REST APIs will significantly enhance scalability, multi-user capabilities, and collaborative potential. Such architectural evolution will enable more efficient use of computational resources, broader access, and robust support for collaborative research initiatives, further solidifying the app as a cornerstone platform in the FAIR simulation landscape.

The QE app represents a significant advancement in making DFT simulations more accessible to the broader communities by offering turnkey solutions that simplify the complexities of computational materials science. Through its intuitive GUI, the app guides users from structure selection to results analysis, reducing barriers for non-specialists. While not intended to replace proper training, it offers a scientifically sound entry point supported by initiatives, such as our dedicated training sessions<sup>112</sup>, helping non-specialists learn through guided practice while gradually building expertise. Its modular, plugin-based architecture supports diverse materials science workflows and allows for the rapid integration of new functionalities, keeping the app up to date with ongoing developments in computational materials science. In conclusion, the QE app lowers the barrier to advanced DFT simulations, enabling a broader range of researchers to adopt computational tools and workflows, accelerating collaboration and driving forward materials research.

## Methods

### Technical details

Given the complexity of the QE app's UI and underlying logic, careful architectural choices are essential to ensure long-term maintainability and extensibility. To this end, the app adopts the Model-View-Controller<sup>113,114</sup> (MVC) design pattern, which provides a clear separation between the UI, the underlying data models, and the computational logic. This structure enhances scalability, simplifies testing, and improves responsiveness by enabling lazy-loading techniques. The details of the MVC implementation, including the use of the Observer and Mediator supporting design patterns via the `traitlets` package, and the model network architecture, are provided in the **Model-View-Controller** section of the Supplementary Information.

### Computational details

All DFT calculations in this work are performed using the default settings of the `aiida-quantumespresso` balanced protocols, as defined in ref. 62. The SSSP v1.3.0 efficiency pseudopotential family<sup>63,115</sup> is used throughout. For the phonon simulations of BaZrS<sub>3</sub> shown in Fig. 4, a 2 × 2 × 2 supercell is employed to compute the force constants, which are necessary to obtain the phonon dispersion and related vibrational properties.

### Code availability

The QE app is available as an open-source project and can be accessed from its GitHub repository: <https://github.com/aiidalab/aiidalab-qe>. The `aiida-quantumespresso` plugin and its associated workflows are hosted at <https://github.com/aiidateam/aiida-quantumespresso>. The AiiDALab platform, which serves as the foundation for these applications, is maintained under the AiiDALab GitHub organization: <https://github.com/aiidalab>. The `aiidalab-launch` tool is hosted at <https://github.com/aiidalab/aiidalab-launch>. Additionally, QE app plugins for different functionalities are available: • `aiidalab-qe-vibrospectroscopy`: <https://github.com/aiidalab/aiidalab-qe-vibrospectroscopy> • `aiidalab-qe-muon`: <https://github.com/aiidalab/aiidalab-qe-muon> • `aiidalab-qe-hp`: <https://github.com/aiidalab/aiidalab-qe-hp> • `aiidalab-qe-wannier90`: <https://github.com/aiidalab/aiidalab-qe-wannier90> • `aiida-qe-xspec`: <https://github.com/aiidaplugins/aiida-qe-xspec> • `aiidalab-qe-pp`: <https://github.com/AndresOrtegaGuerrero/aiidalab-qe-pp> • `aiida-bader`: <https://github.com/superstar54/aiida-bader>. All software mentioned in this paper is open-source and freely available to the community.

Received: 25 July 2025; Accepted: 15 December 2025;

Published online: 03 February 2026

## References

- Hohenberg, P. & Kohn, W. Inhomogeneous electron gas. *Phys. Rev.* **136**, B864–B871 (1964).
- Kohn, W. & Sham, L. J. Self-consistent equations including exchange and correlation effects. *Phys. Rev.* **140**, A1133 (1965).
- Marzari, N., Ferretti, A. & Wolverton, C. Electronic-structure methods for materials design. *Nat. Mater.* **20**, 736–749 (2021).
- Talirz, L., Ghiringhelli, L. M. & Smit, B. Trends in atomistic simulation software usage [article v1.0]. *Living J. Comput. Mol. Sci.* **3**, 1483 (2021).
- Lejaeghere, K. et al. Reproducibility in density functional theory calculations of solids. *Science* **351**, aad3000 (2016).
- Bosoni, E. et al. How to verify the precision of density-functional-theory implementations via reproducible and universal workflows. *Nat. Rev. Phys.* **6**, 45–58 (2024).
- Blum, V. et al. Roadmap on methods and software for electronic structure based simulations in chemistry and materials. *Electron. Struct.* **6**, 042501 (2024).
- Jain, A. et al. Commentary: the materials project: a materials genome approach to accelerating materials innovation. *APL Mater.* **1**, 011002 (2013).
- Curtarolo, S. et al. The high-throughput highway to computational materials design. *Nat. Mater.* **12**, 191–201 (2013).

10. Cheon, G. et al. Data mining for new two- and one-dimensional weakly bonded solids and lattice-commensurate heterostructures. *Nano Lett.* **17**, 1915–1923 (2017).
11. Mounet, N. et al. Two-dimensional materials from high-throughput computational exfoliation of experimentally known compounds. *Nat. Nanotechnol.* **13**, 246–252 (2018).
12. Mounet, N. et al. Two-dimensional materials from high-throughput computational exfoliation of experimentally known compounds. *Mater. Cloud Archive* **157**, <https://doi.org/10.24435/materialscloud:yf-kf> (2024).
13. Vecchio, K. S., Dippo, O. F., Kaufmann, K. R. & Liu, X. High-throughput rapid experimental alloy development (HT-READ). *Acta Materialia* **221**, 117352 (2021).
14. Gjerding, M. N. et al. Recent progress of the computational 2D materials database (C2DB). *2D Mater.* **8**, 044002 (2021).
15. Huber, S. et al. Materials Cloud three-dimensional crystals database (MC3D). *Materials Cloud Archive* **38**, <https://doi.org/10.24435/materialscloud:rw-t0> (2022).
16. Curtarolo, S. et al. AFLOW: An automatic framework for high-throughput materials discovery. *Comput. Mater. Sci.* **58**, 218–226 (2012).
17. Pizzi, G., Cepellotti, A., Sabatini, R., Marzari, N. & Kozinsky, B. AiiDA: automated interactive infrastructure and database for computational science. *Comput. Mater. Sci.* **111**, 218–230 (2016).
18. Uhrin, M., Huber, S. P., Yu, J., Marzari, N. & Pizzi, G. Workflows in AiiDA: engineering a high-throughput, event-based engine for robust and modular computational workflows. *Comput. Mater. Sci.* **187**, 110086 (2021).
19. Huber, S. P. et al. AiiDA 1.0, a scalable computational infrastructure for automated reproducible workflows and data provenance. *Sci. Data* **7**, 300 (2020).
20. Gjerding, M. et al. Atomic Simulation Recipes: A Python framework and library for automated workflows. *Comput. Mater. Sci.* **199**, 110731 (2021).
21. Mortensen, J. J., Gjerding, M. & Thygesen, K. S. MyQueue: task and workflow scheduling system. *J. Open Source Softw.* **5**, 1844 (2020).
22. Mathew, K. et al. Atomate: A high-level interface to generate, execute, and analyze computational materials science workflows. *Comput. Mater. Sci.* **139**, 140–152 (2017).
23. Jain, A. et al. Fireworks: a dynamic workflow system designed for high-throughput applications. *Concurrency Comput. Pract. Exp.* **27**, 5037–5059 (2015).
24. Cunningham, W. et al. Agnostiqhq/covalent: v0.240.0 <https://doi.org/10.5281/zenodo.15400489> (2025).
25. Rosen, A. S. et al. Jobflow: computational workflows made simple. *J. Open Source Softw.* **9**, 5995 (2024).
26. Armiento, R. Database-Driven High-Throughput Calculations and Machine Learning Models for Materials Design. In Schütt, K. T. et al. (eds.) *Machine Learning Meets Quantum Physics*, 377–395 [https://doi.org/10.1007/978-3-030-40245-7\\_17](https://doi.org/10.1007/978-3-030-40245-7_17) (Springer International Publishing, 2020).
27. Atwi, R., Bliss, M., Makeev, M. & Rajput, N. N. MISPR: an open-source package for high-throughput multiscale molecular simulations. *Sci. Rep.* **12**, 15760 (2022).
28. Janssen, J. et al. pyiron: an integrated development environment for computational materials science. *Comput. Mater. Sci.* **163**, 24–36 (2019).
29. Kirklin, S. et al. The open quantum materials database (oqmd): assessing the accuracy of dft formation energies. *npj Comput. Mater.* **1**, 1–15 (2015).
30. Wilkinson, M. D. et al. The FAIR guiding principles for scientific data management and stewardship. *Sci. Data* **3**, 1–9 (2016).
31. Huber, S. P. et al. Common workflows for computing material properties using different quantum engines. *npj Comput. Mater.* **7**, 136 (2021).
32. Ganose, A. M. et al. Atomate2: Modular Workflows for Materials Science. *ChemRxiv* <https://doi.org/10.26434/chemrxiv-2025-tcr5h> (2025).
33. Rosen, A. quacc - The Quantum Accelerator <https://doi.org/10.5281/zenodo.15641940> (2025).
34. Baerends, E. J. et al. The amsterdam modeling suite. *J. Chem. Phys.* **162**, 162501 (2025).
35. Marchesin, F. et al. Atomistic simulation advanced platform (asap) for materials modelling with *ab initio* methods. In *Psi-k Conference 2022 Abstract Book* (ASAP, 2022).
36. mat3ra. <https://www.mat3ra.com/> (MAT3RA, 2025).
37. Meunier, M. & Robertson, S. Materials studio 20th anniversary (Materials Studio, 2021).
38. Materys. <https://www.materys.com> (Materys, 2025).
39. Medea. <https://www.materialsdesign.com/medea-software> (Medea, 2025).
40. Smidstrup, S. et al. Quantumatk: an integrated platform of electronic and atomic-scale modelling tools. *J. Phys. Condens. Matter* **32**, 015901 (2019).
41. Schrödinger. <https://www.schrodinger.com> (Schrodinger, 2025).
42. Rêgo, C. R. et al. SimStack: an intuitive workflow framework. *Front. Mater.* **9**, 877597 (2022).
43. Kokalj, A., Sbraccia, C. & Giannozzi, P. PWgui. <http://www-k3.ijs.si/kokalj/pwgui/pwgui.html> (PWGUI, 2014).
44. Burai. <https://github.com/BURAI-team/burai> (2025).
45. Mortensen, J. J. et al. GPAW: an open Python package for electronic structure calculations. *J. Chem. Phys.* **160**, 092503 (2024).
46. Gupta, K., Bhattacharjee, S. & Lee, S.-C. CINEMAS: comprehensively integrated environment for advanced materials simulations. *Comput. Mater. Sci.* **188**, 110238 (2021).
47. Wang, G. et al. ALKEMIE: an intelligent computational platform for accelerating materials discovery and design. *Comput. Mater. Sci.* **186**, 110064 (2021).
48. Yakutovich, A. V. et al. AiiDALab - an ecosystem for developing, executing, and sharing scientific workflows. *Comput. Mater. Sci.* **188**, 110165 (2021).
49. Giannozzi, P. et al. Quantum ESPRESSO: a modular and open-source software project for quantum simulations of materials. *J. Phys. Condens. Matter* **21**, 395502 (2009).
50. Giannozzi, P. et al. Advanced capabilities for materials modelling with Quantum ESPRESSO. *J. Phys. Condens. Matter* **29**, 465901 (2017).
51. Giannozzi, P. et al. Quantum ESPRESSO toward the exascale. *J. Chem. Phys.* **152**, 154105 (2020).
52. Appmode: a Jupyter extension that turns notebooks into web applications. <https://github.com/oschuett/appmode> (Oschuett, 2025).
53. Goel, A. *Computer fundamentals* (Pearson Education India, 2010).
54. Crusoe, M. R. et al. Methods included: standardizing computational reuse and portability with the common workflow language. *Commun. ACM* **65**, 54–63 (2022).
55. Griem, L. et al. KadiStudio: FAIR modelling of scientific research processes. *Data Sci. J.* **21**, 16–16 (2022).
56. Andersen, C. W. et al. OPTIMADE, an API for exchanging materials data. *Sci. Data* **8**, 217 (2021).
57. Evans, M. L. et al. Developments and applications of the optimade api for materials discovery, design, and data exchange. *Digit. Discov.* **3**, 1509–1533 (2024).
58. Talirz, L. et al. Materials cloud, a platform for open computational science. *Sci. Data* **7**, 299 (2020).
59. Schmidt, J., Pettersson, L., Verdozzi, C., Botti, S. & Marques, M. A. L. Crystal graph attention networks for the prediction of stable materials. *Sci. Adv.* **7**, eabi7948 (2021).
60. Schmidt, J. et al. Machine-learning-assisted determination of the global zero-temperature phase diagram of materials. *Adv. Mater.* **35**, 2210788 (2023).
61. Mosquera-Lois, I., Kavanagh, S. R., Walsh, A. & Scanlon, D. O. ShakeNBreak: navigating the defect configurational landscape. *J. Open Source Softw.* **7**, 4817 (2022).

62. de Miranda Nascimento, G. et al. Accurate and efficient protocols for high-throughput first-principles materials simulations. *arXiv* **2504.03962** (2025).
63. Prandini, G., Marrazzo, A., Castelli, I. E., Mounet, N. & Marzari, N. Precision and efficiency in solid-state pseudopotential calculations. *npj Comput. Mater.* **4**, 72 (2018).
64. van Setten, M. et al. The PseudoDojo: training and grading a 85 element optimized norm-conserving pseudopotential table. *Computer Phys. Commun.* **226**, 39–54 (2018).
65. Grosso, G. & Parravicini, G. P. Solid State Physics, second edn, <https://www.sciencedirect.com/science/article/pii/B9780123850300000050> (Academic Press, 2014).
66. Hinuma, Y., Pizzi, G., Kumagai, Y., Oba, F. & Tanaka, I. Band structure diagram paths based on crystallography. *Comput. Mater. Sci.* **128**, 140–184 (2017).
67. Togo, A., Shinohara, K. & Tanaka, I. Spglib: a software library for crystal symmetry search. *Sci. Technol. Adv. Mater. Methods* **4**, 2384822 (2024).
68. Schuler, B. et al. How substitutional point defects in two-dimensional WS<sub>2</sub> induce charge localization, spin-orbit splitting, and strain. *ACS Nano* **13**, 10520–10534 (2019).
69. Nielsen, R. S. et al. BaZrS<sub>3</sub> lights up: The interplay of electrons, photons, and phonons in strongly luminescent single crystals. *Adv. Opt. Mater.* **13**, e00915 (2025).
70. Bastonero, L. & Marzari, N. Automated all-functionals infrared and Raman spectra. *npj Comput. Mater.* **10**, 55 (2024).
71. Bastonero, L. aiida-phonopy. <https://github.com/aiida-phonopy/aiida-phonopy> (2025).
72. Togo, A., Chaput, L., Tadano, T. & Tanaka, I. Implementation strategies in phonopy and phono3py. *J. Phys. Condens. Matter* **35**, 353001 (2023).
73. Togo, A. First-principles phonon calculations with phonopy and phono3py. *J. Phys. Soc. Jpn.* **92**, 012001 (2023).
74. Baroni, S., de Gironcoli, S., Dal Corso, A. & Giannozzi, P. Phonons and related crystal properties from density-functional perturbation theory. *Rev. Mod. Phys.* **73**, 515–562 (2001).
75. Umari, P. & Pasquarello, A. Ab initio Molecular Dynamics in a Finite Homogeneous Electric Field. *Phys. Rev. Lett.* **89**, 157602 (2002).
76. Fair, R. et al. *Euphonic*: inelastic neutron scattering simulations from force constants and visualization tools for phonon properties. *J. Appl. Crystallogr.* **55**, 1689–1703 (2022).
77. van Bokhoven, J. A. & Lamberti, C. *X-Ray Absorption and X-Ray Emission Spectroscopy*, vol. 1–2 <https://doi.org/10.1002/9781118844243> (Wiley, 2016).
78. Gougoussis, C., Calandra, M., Seitsonen, A. P. & Mauri, F. First-principles calculations of x-ray absorption in a scheme based on ultrasoft pseudopotentials: From  $\alpha$ -quartz to high- $T_c$  compounds. *Phys. Rev. B* **80**, 075102 (2009).
79. Bunäu, O. & Calandra, M. Projector augmented wave calculation of x-ray absorption spectra at the  $L_{2,3}$  edges. *Phys. Rev. B* **87**, 205105 (2013).
80. Triguero, L., Pettersson, L. & Ågren, H. Calculations of near-edge x-ray-absorption spectra of gas-phase and chemisorbed molecules by means of density-functional and transition-potential theory. *Phys. Rev. B* **58**, 8097 (1998).
81. Cavgliasso, G. & Chong, D. P. Accurate density-functional calculation of core-electron binding energies by a total-energy difference approach. *J. Chem. Phys.* **111**, 9485–9492 (1999).
82. Walter, M., Moseler, M. & Pastewka, L. Offset-corrected  $\delta$ -Kohn-Sham scheme for semiempirical prediction of absolute X-ray photoelectron energies in molecules and solids. *Phys. Rev. B* **94**, 041112 (2016).
83. de Réotier, P. D. & Yaouanc, A. Muon spin rotation and relaxation in magnetic materials. *J. Phys. Condens. Matter* **9**, 9113 (1997).
84. Möller, J. S., Ceresoli, D., Lancaster, T., Marzari, N. & Blundell, S. J. Quantum states of muons in fluorides. *Phys. Rev. B* **87**, 121108 (2013).
85. Bernardini, F., Bonfà, P., Massidda, S. & De Renzi, R. Ab initio strategy for muon site assignment in wide band gap fluorides. *Phys. Rev. B* **87**, 115148 (2013).
86. Blundell, S. J. et al. Electronic structure calculations for muon spectroscopy. *Electron. Struct.* **7**, 023001 (2025).
87. Onuorah, I. J. et al. Automated computational workflows for muon spin spectroscopy. *Digital Discov.* **4**, 523–538 (2025).
88. Onuorah, I. J. et al. Positivemuon/aiida-muon: v1.0.3, <https://doi.org/10.5281/zenodo.14594493> (2025).
89. Bonfà, P., Frassinetti, J., Isah, M. M., Onuorah, I. J. & Sanna, S. UNDI: an open-source library to simulate muon-nuclear interactions in solids. *Computer Phys. Commun.* **260**, 107719 (2021).
90. Marzari, N. & Vanderbilt, D. Maximally localized generalized Wannier functions for composite energy bands. *Phys. Rev. B* **56**, 12847 (1997).
91. Marzari, N., Mostofi, A. A., Yates, J. R., Souza, I. & Vanderbilt, D. Maximally localized Wannier functions: theory and applications. *Rev. Mod. Phys.* **84**, 1419–1475 (2012).
92. Marrazzo, A. et al. Wannier-function software ecosystem for materials simulations. *Rev. Mod. Phys.* **96**, 045008 (2024).
93. Damle, A., Lin, L. & Ying, L. Compressed representation of Kohn–Sham orbitals via selected columns of the density matrix. *J. Chem. Theory Comput.* **11**, 1463–1469 (2015).
94. Vitale, V. et al. Automated high-throughput Wannierisation. *npj Comput. Mater.* **6**, 66 (2020).
95. Qiao, J., Pizzi, G. & Marzari, N. Projectability disentanglement for accurate and automated electronic-structure Hamiltonians. *npj Comput. Mater.* **9**, 208 (2023).
96. Biberacher, W. Shubnikov-de Haas and de Haas-van Alphen Techniques. *Encycl. Condens. Matter Phys.* **4**, 228–236 (2005).
97. Bergemann, C. Fermi Surface Measurements. *Encyclopedia of Condensed Matter Physics* 185–192 (Elsevier, 2005).
98. Anisimov, V., Zaanen, J. & Andersen, O. Band theory and Mott insulators: Hubbard  $U$  instead of Stoner  $I$ . *Phys. Rev. B* **44**, 943 (1991).
99. Liechtenstein, A., Anisimov, V. & Zaanen, J. Density-functional theory and strong interactions: orbital ordering in Mott-Hubbard insulators. *Phys. Rev. B* **52**, R5467 (1995).
100. Dudarev, S., Botton, G., Savrasov, S., Humphreys, C. & Sutton, A. Electron-energy-loss spectra and the structural stability of nickel oxide: an LSDA+ $U$  study. *Phys. Rev. B* **57**, 1505 (1998).
101. Campo, V. L. & Cococcioni, M. Extended DFT+ $U$ + $V$  method with on-site and inter-site electronic interactions. *J. Phys. Condens. Matter* **22**, 055602 (2010).
102. Bastonero, L. et al. First-principles Hubbard parameters with automated and reproducible workflows. *npj Comput. Mater.* **11**, 183 (2025).
103. Timrov, I., Marzari, N. & Cococcioni, M. Hubbard parameters from density-functional perturbation theory. *Phys. Rev. B* **98**, 085127 (2018).
104. Timrov, I., Marzari, N. & Cococcioni, M. Self-consistent Hubbard parameters from density-functional perturbation theory in the ultrasoft and projector-augmented wave formulations. *Phys. Rev. B* **103**, 045141 (2021).
105. Timrov, I., Marzari, N. & Cococcioni, M. HP-A code for the calculation of Hubbard parameters using density-functional perturbation theory. *Computer Phys. Commun.* **279**, 108455 (2022).
106. Otero-de-la-Roza, A., Johnson, E. R. & Luaña, V. Critic2: a program for real-space analysis of quantum chemical interactions in solids. *Computer Phys. Commun.* **185**, 1007–1018 (2014).
107. Wang, X. Weas widget. <https://github.com/superstar54/weas-widget> (Weas, 2025).
108. Tang, W., Sanville, E. & Henkelman, G. A grid-based Bader analysis algorithm without lattice bias. *J. Phys. Condens. Matter* **21**, 084204 (2009).
109. aiidalab-empa-surfaces. <https://github.com/nanotech-empa/aiidalab-empa-surfaces> (2025).

110. Du, D., Baird, T. J., Bonella, S. & Pizzi, G. OSSCAR, an open platform for collaborative development of computational tools for education in science. *Computer Phys. Commun.* **282**, 108546 (2023).
111. Du, D., Baird, T. J., Eimre, K., Bonella, S. & Pizzi, G. Jupyter widgets and extensions for education and research in computational physics and chemistry. *Computer Phys. Commun.* **305**, 109353 (2024).
112. PSI Laboratory for Materials Simulations. Course “Electronic-structure simulations for user communities at large-scale facilities”, 3–9 April 2025, Villigen PSI, Switzerland. <https://indico.psi.ch/event/17436/> (2025).
113. Krasner, G. E. & Pope, S. T. A description of the model-view-controller user interface paradigm in the Smalltalk-80 system. *J. Object-Oriented Program.* **1**, 26–49 (1988).
114. Gamma, E., Helm, R., Johnson, R. & Vlissides, J. *Design Patterns: Elements of Reusable Object-Oriented Software* (Addison-Wesley, 1994).
115. Prandini, G. et al. A Standard Solid State Pseudopotentials (SSSP) library optimized for precision and efficiency. *Materials Cloud Archive* **2023.65** <https://doi.org/10.24435/materialscloud:f3-ym> (2023).
116. Perdew, J. P., Burke, K. & Ernzerhof, M. Generalized gradient approximation made simple. *Phys. Rev. Lett.* **77**, 3865–3868 (1996).
117. Bobzien, L. et al. Layer-Dependent charge-state lifetime of single Se vacancies in  $\text{WSe}_2$ . *Phys. Rev. Lett.* **134**, 076201 (2025).

## Acknowledgements

We gratefully thank Carl Simon Adorf for initial contributions and developments to the QE app. We thank the AiiDA team for their continuous support and contributions to the AiiDA framework, which underpins the QE app, the QE developers for their ongoing efforts in maintaining and improving the QE code, which is the computational engine behind the app, and all users who tested the app and provided valuable feedback, including Nicola Colonna, Thomas J. Hicken, Jonas A. Krieger, Stanislav Nikitin and Tom Fennell. XW, EB, MBo, AOG, MBe, DD, SPH, NP, JQ, TR, CJS, IT, AVY, JY, NM, CAP and GP acknowledge financial support by the NCCR MARVEL, a National Center of Competence in Research, funded by the Swiss National Science Foundation (grant number 205602). EB, CAP and GP acknowledge financial support by the Open Research Data Program of the ETH Board (project “PREMISE”: Open and Reproducible Materials Science Research). EM, DP and NM acknowledge support by the MaX European Center of Excellence – Materials design at the eXascale ([www.max-centre.eu](http://www.max-centre.eu)), of which QE is a lighthouse code; MaX is supported by the European High Performance Computing Joint Undertaking and participating countries (grant No. 101093374). PNOG and DP acknowledge financial support by the European Union - NextGenerationEU through the Italian Ministry of University and Research under PNRR - M4C2I1.4 ICSC - Centro Nazionale di Ricerca in High Performance Computing, Big Data and Quantum Computing (Grant No. CN0000013) through the Innovation Grant ASGARD. LB and NM acknowledge financial support by the Deutsche Forschungsgemeinschaft (DFG) under Germany’s Excellence Strategy (EXC 2077, No. 390741603, University Allowance, University of Bremen) and Lucio Colombi Ciacchi, the host of the “U Bremen Excellence Chair Program”. NP, JY and GP acknowledge financial support by the Swiss National Science Foundation (SNSF) Project Funding (grant 200021E\_206190 “FISH4DIET”). IT acknowledges financial support by the Swiss National Science Foundation (SNSF) Project Funding (grant 200021-227641 and 200021-236507). MBe, XW, JY and GP acknowledge financial support by the SwissTwins project, funded by the Swiss State Secretariat for Education, Research and Innovation (SERI). DD and GP acknowledge financial support by the EPFL Open Science Fund via the OSSCAR project. XW, PNOG, MAHB, EM, DP, NM and GP acknowledge financial support by the European Union’s Horizon 2020 research and innovation program under grant agreement No. 957189 (BIG-MAP), also part of the BATTERY 2030+ initiative under grant agreement No. 957213. JY, NM and GP acknowledge financial support by the MARKETPLACE

project funded by Horizon 2020 under the H2020-NMBP-25-2017 call (Grant No. 760173). DH acknowledges financial support by the European Union’s Horizon 2020 research and innovation program under grant agreement No. 803718 (SINDAM) and the UK Research and Innovation (UKRI) EPSRC grant ref. EP/X026973/1. PB, RDR and IJO acknowledge financial support by the PNRR MUR project ECS-0000033-ECOSISTER and from University of Parma through the action “Bando di Ateneo 2023 per la ricerca”. We acknowledge access to Piz Daint and Alps at the Swiss National Supercomputing Center (CSCS), Switzerland under the MARVEL’s share with the project IDs mr32 and mr33. This work was further supported by grants from the Swiss National Supercomputing Center (CSCS) under project IDs lp18, s1267 and s1295 on Alps. We acknowledge ISCRA, ECOSISTER and ICSC for awarding access to the LEONARDO supercomputer, hosted by CINECA (Italy).

## Author contributions

Conceptualization: X.W., E.B., M.Bo., A.O.G., A.V.Y., J.Y., N.M., C.A.P., and G.P. Methodology: X.W., E.B., M.Bo., A.O.G., L.B., M.Be., P.B., R.D.R., D.D., P.N.O.G., M.A.H.B., D.H., S.P.H., E.M., I.J.O., N.P., D.P., J.Q., T.R., C.J.S., I.T., A.V.Y., J.Y., N.M., C.A.P., and G.P. Software: X.W., E.B., M.Bo., A.O.G., L.B., M.Be., P.B., D.D., P.N.O.G., M.A.H.B., D.H., S.P.H., I.J.O., N.P., D.P., J.Q., T.R., C.J.S., I.T., A.V.Y., J.Y., C.A.P., and G.P. Validation: X.W., E.B., M.Bo., A.O.G., D.P., J.Y., C.A.P., and G.P. Investigation: X.W., E.B., M.Bo., A.O.G., and J.Y. Resources: R.D.R., E.M., D.P., N.M., C.A.P., and G.P. Writing - original draft: X.W., E.B., M.Bo., and A.O.G. Writing - review & editing: X.W., E.B., M.Bo., A.O.G., L.B., M.Be., P.B., R.D.R., P.N.O.G., M.A.H.B., D.H., E.M., I.J.O., N.P., D.P., J.Q., T.R., I.T., A.V.Y., J.Y., N.M., C.A.P., and G.P. Visualization: X.W., E.B., M.Bo., A.O.G., P.N.O.G., M.A.H.B. T.R., A.V.Y., and J.Y. Supervision: R.D.R., E.M., D.P., N.M., C.A.P., and G.P. Project administration: N.M., C.A.P., and G.P. Funding acquisition: R.D.R., E.M., D.P., N.M., C.A.P., and G.P.

## Competing interests

The authors declare no competing interests.

## Additional information

**Supplementary information** The online version contains supplementary material available at <https://doi.org/10.1038/s41524-025-01936-4>.

**Correspondence** and requests for materials should be addressed to Xing Wang, Carlo A. Pignedoli or Giovanni Pizzi.

**Reprints and permissions information** is available at <http://www.nature.com/reprints>

**Publisher’s note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2026