English Language Remote Workshop
IBExpert, Firebird, AI and PHP web applications
4th-5th May 2026, 09:00-17:00 (Central European Time)

Many software providers have been successful for years with their own industry software in typical workgroup installations at end customers. Sooner or later, however, the same question almost always arises: How can existing applications be meaningfully expanded for mobile use?

Today, there is a demand for web-based modules for bookings, feedback, time logging, approvals or simple data entry directly from the central system. This information should not only be able to be displayed on a mobile phone or tablet, but ideally also be written back directly to the central database system via a simple, intuitive interface. At the same time, it must of course be ensured that no new security gaps are created unintentionally and that your own data does not become unsecured and available on the internet.

This is precisely where many projects fail in practice. What may appear to be a simple extension at first glance quickly proves to be a technically, organisationally and human resource-intensive task.

A typical example is mobile time-sheet recording: users should not only be able to start and stop recording on their smartphones, but also view information such as target hours, overtime or shortfall hours from the central database. In addition, it should be possible to enter text to document activities. What seems technically simple usually required a team of web developers, web server administrators and application developers, who had to work together to construct a completely new technical environment. Such projects often failed at this early stage, or later became so complex that they were hardly economically viable.

There is another point to consider: many companies are still trying to implement such requirements with native apps for iOS or Android, often based on Delphi or other source code tools. Even with manageable requirements, this quickly leads to enormous efforts between the idea and its functional implementation. What initially looks like a clean solution often develops into a project lasting many months or even several years. On top of that, there are problems with libraries, dependencies, version statuses and security gaps. Libraries that seemed ideal at the start of the project are eventually no longer maintained or no longer fit new platform requirements.

The distribution of native applications via app stores is also a factor that should not be underestimated in practice. Different platform guidelines, approval processes, technical restrictions and problems with older devices regularly mean that even the installation of the application no longer works reliably. Many projects start with big ideas, but then fail due to massive time and budget overruns and are ultimately never completed. It is not uncommon for the employees originally responsible for these projects to have long since left the company by this point.

At the same time, it is becoming increasingly difficult to find new employees who understand both the company's own industry software and have genuine experience in the successful implementation of mobile applications. Qualified specialists in this field are rare and expensive, and affordable young talent is often not available to the extent that companies might need it.

Our workshop approach

This is precisely where our workshop comes to the fore.

As the owner of IBExpert, Holger Klemt has been actively involved in customer projects for many years, where not only in-depth database knowledge is required, but above all the ability to mediate between the technical requirements of a company and a practical technical implementation. It is therefore not just a matter of understanding databases, but of designing applications in such a way that they are convenient, flexible and maintainable for users in the long term — without having to start all over again every time a change is requested.

Many of our customer projects run for ten, fifteen or twenty years and must be continuously adapted to new requirements during this time. The platforms of choice are often Firebird and Lazarus or Delphi. These tools are extremely powerful, but not exactly renowned for their ability to create mobile applications for iOS or Android on the fly.

For many years, we have been using PHP to fill this gap as a pragmatic, stable and extremely flexible way of bringing existing systems to the web in a meaningful way.

A cleanly configured web server is technically not rocket science. In practice, however, even seemingly minor issues such as SSL certificates, server setup, PHP modules or firewall adjustments often take up more time than you would like to spend on them in your day-to-day project work. That is why we deliberately take a different approach in the workshop: we start with a practical approach from scratch.

Begin with an empty cloud server

At the beginning of the workshop, we work with a completely empty cloud server, which we set up live. We use a server from Hetzner as an example. Such systems are available at very reasonable monthly costs and still offer sufficient performance for serious applications. This lowers the entry barrier, and many participants realise during the workshop that such a setup can also be used economically for their own projects.

We use a up-to-date Ubuntu operating system with full root access and a public IP address. It is already clear at this point that we are not working in an artificial demo environment, but in a realistic technical infrastructure that can actually be used productively later on.

The next step is to set up the server under a subdomain. Alternatively, you can directly use your own domain. This is followed by the basic installation steps: Apache, PHP, php_firebird and the other components necessary for a functional web environment.

This already highlights a key advantage of the workshop concept: many of these steps are prepared and expedited with the support of AI, in particular ChatGPT. In IT, the principle has always been that you don't have to know everything yourself as long as you can reliably access the right information. This is precisely where ChatGPT has proven to be an extremely helpful tool — especially when generating PHP applications, installation steps, scripts and configuration aids.

Instead of laboriously figuring out each individual shell command ourselves, we use AI specifically as an assistance system. Anyone who has ever manually made an Apache web server SSL-enabled will quickly recognise how much time and effort can be saved by clearly formulated instructions and the scripts

generated from them. After just a short time, you will have a functional, secure web environment that can be used to run the first PHP tests directly.

Firebird and the first practical test

Since our focus is on high-performance access from PHP to Firebird, the next step is to install Firebird 5 on the Linux server. Here, too, the necessary configuration steps are carried out in a practical way, including adjustments to the firebird.conf, other system-relevant settings and firewall clearance.

Once the environment is set up, we create a test database on the server and then open it with an externally running version of IBExpert. There, we create the first tables and run basic function tests.

This is followed by one of the most exciting demonstrations of the entire workshop for many participants: with just a few well-formulated instructions to ChatGPT, we generate a simple SQL web interface. After entering the connection string, user name and password, any SQL statements can be executed on the database and the results displayed directly in the browser.

A well-formulated block of instructions consisting of just a few lines is often sufficient to obtain a functional index.php in no time at all. This is transferred to the server via SFTP and run immediately.

Participants experience first-hand that all the essential steps leading up to the first executable web application can be successfully completed on the first day, and usually before the first lunch break. More importantly, the result can be utilised immediately and is accessible worldwide.

After lunch: from technical setup to real-world application

Once the technical basis has been established, the specialised part follows.

Together, we develop a suitable table model for the specific applications discussed in the workshop. Based on this, a data model is created and implemented on the database server using test data.

From this point onwards, we begin with the steps that are of particular interest to many technical departments, software architects and project managers. This is because the focus is no longer primarily on servers, services and configurations, but on the actual application.

Based on the data model, we define the requirements for input forms, views, analyses and processes. We determine which data should be displayed where, which inputs must be possible and how users should be guided through the application.

The crucial point here is that these requirements are not formulated in the form of complicated or technical specifications, but are described in natural English. It is precisely this description that is then passed on to ChatGPT as a work assignment.

This fundamentally changes the nature of development. Instead of having to write every single line of PHP, HTML, CSS or JavaScript yourself, you can concentrate on the technical idea and a clear description of what the application should deliver. The AI quickly delivers executable implementations that can be tested, improved and expanded directly.

Experience shows that visible progress can be achieved in an astonishingly short time using this method. Your own expertise is not replaced, but optimally supplemented. If a generated version is still unusable or incomplete, you can immediately report the test results and usually receive an improved version shortly afterwards.

This iterative collaboration is one of the greatest practical advantages. You can immediately see how quickly an idea can be turned into an attractive and functional web application that can be run directly on a physical online server.

Whether this application is then used with a modern smartphone, an older iPad or a Windows PC is initially irrelevant. The solution is immediately accessible cross-platform. Of course, in individual cases, you will find that certain devices or platforms — especially Apple iOS — have peculiarities that need to be taken into account. But precisely these points can be analysed directly, and specifically adapted with the help of ChatGPT.

Anyone who has ever seen how a clearly formulated description can be turned into functional source code within a short period of time will quickly realise that the focus of development is shifting: away from detailed knowledge of CSS attributes, HTML structures or JavaScript special cases — and towards the actual idea, the process and the technically effective design of the application.

Day 2: Connection to live systems

The second day focuses on integration into practice.

Now the web application is connected to a real customer system. The underlying database platform is by no means limited to Firebird 5. Other systems that can be accessed via Lazarus SQLDB or similar mechanisms can also be integrated — such as MSSQL, PostgreSQL, Oracle, ODBC data sources or, in simple cases, even text-based databases.

Similarly, the approach shown is not limited to Lazarus or Delphi. Existing source codes from other platforms such as C# can also be integrated into such architectures.

Technically, we rely on a push/pull principle. The web server is already accessible on the internet with a fixed public IP address. At the same time, access via the encrypted Firebird protocol is also possible if required.

The basic idea is to transfer all data records relevant to the web application from the end customer's database, which is typically located in the internal network behind a firewall, to the web server in a targeted manner. If new entries or new data records are then created in the web application, these are first stored centrally in the database there.

A local client in the end customer's network then recognises these changes asynchronously and almost in real time. In the Firebird environment, this can be achieved, for example, using event alert-controlled processes. In this way, newly created data can be recognised in a fraction of a second and automatically transferred back to the local target system.

Another application is used for the return path. In the workshop, we demonstrate this using, among other things, an IBEBLOCK script in conjunction with the IBExpert module ibescript.exe, which can also be passed on to your own customers as part of the IBExpert Server Tools or the IBExpert OEM solutions.

Extension, user management and API

In further steps, new functions will be added to the web interface. Here, too, we will again use specific instructions to ChatGPT to implement enhancements quickly.

This includes, for example, user login, identification of individual users, role-based usage, or the basis for server-side APIs. This not only allows the application to be opened up for direct user access, but also, if necessary, to be expanded so that other software providers or third-party systems can access it in a controlled manner.

In this way, a simple web mask is gradually transformed into a complete, practical web application that integrates seamlessly into existing software landscapes.

Target group for the workshop

The workshop is aimed at several target groups.

On the one hand, experienced Delphi, Lazarus or C# developers who have little or no practical experience with the real-world implementation of mobile web applications for existing database systems, or who have had negative experiences with traditional approaches.

On the other hand, the workshop is also aimed at business software experts, project managers and management teams who can describe in great detail what a mobile application should do for their software, but who lack the human or technical resources in-house to implement such a solution economically and practically.

Conclusion

The workshop demonstrates a practical approach to transforming existing database applications into modern mobile web solutions in a remarkably short time using Firebird, IBExpert, PHP and AI support.

Instead of months of app development, complex toolchains and special projects that are difficult to calculate, you learn an approach that quickly leads to real results, remains technically manageable and can be directly transferred to your own projects.

The focus is not on theoretical concepts, but on concrete, executable results: from empty cloud servers and the first SSL-enabled web environment to functioning web applications and their connection to real customer systems.

Secure your place

Price per person: EUR 899 (net of any statutory VAT/sales tax)
Registration: sales@ibexpert.biz

Can't make it on this date? We also offer the workshop individually as a company remote workshop.