

## Access JumpStart

The HCI Access JumpStart (AJS) product provides a feature-rich Microsoft Access application template, consisting of Front End and Back End Access database files.

It “kick starts” the build of a new user application or gives additional capabilities to an existing application.

### Benefits

- Reduces your time and effort to deploy a Microsoft Access application.
- Helps you provide a consistent look and feel to your customer within and between applications.
- Reduces your effort and clock time to enhance delivered Access applications.
- Minimizes your effort to diagnose and repair problems.
- Gives you more time to focus on the business rules, workflow and user interface characteristics unique to each customer situation, for increased customer satisfaction.
- Improves user productivity because each end user has a view of the application tailored to their role, such as Manager, Salesperson, Billing Clerk, etc.
- Allows you to give a level of operational control to customer Administrators, thereby reducing your support workload and improving customer satisfaction.
- The customer experiences reduced Mean Time to Repair (MTTR) of problems encountered.
- You can easily provide auditing messages for customer Managers, etc.

### Licensing

AJS is licensed on a per-seat basis. Each developer must obtain a license to use the product and needs to maintain a subscription for support and to obtain new product releases.

Applications built with a licensed copy of the AJS product may be freely distributed to your end-user customers. There are no end user licensing requirements for such applications.

### Structure

The AJS structure is a “skeleton” application with Microsoft Access Front End and Back End files offering a mature development framework. It also includes a library of support modules to simplify VBA code development.

- **Option parameters**
  - The developer specifies Framework options in one place to control behavior of the AJS environment.
  - Developer-built VBA code can also refer to these parameters.
  - In addition, the Developer can add new application-specific Option Parameters as desired.
- **Back End database**
  - The primary back end database may be an Access file, SQL Server, Azure SQL or MySQL database.
  - Connection is automatically made at application startup and optional User ID and Password fields are stored in encrypted form in the front end template.
  - Additional secondary back end databases may be specified and may be attached and detached independently via VBA calls to the AJS library.

- **Session Manager**
  - The user LoginID determines how each user may interact with the application.
  - At session start the user's defined Role is determined and is used by other components and by developer-written VBA code to control the user application view and experience.
- **Log Manager:**
  - A session log is maintained for each user session with Start and End messages
  - The menu system optionally logs each menu selection in the session.
  - The error processing component logs error information for the session.
  - You can write any desired log status message from VBA with one line of code.
- **Menu System:** The menu system allows multiple menus with any number of item entries per page (limited only by the maximum number of objects on a form). It provides a restricted view of menu items based upon user Role. Forms are provided so that developers and/or user administrators may
  - Set user authorization and
  - Display the most recent login information for each user.
- **Error Processing:** The Error Processing component can provide global error processing for unhandled as well as handled error traps. It is designed to be used with or without the optional [vbWatchdog](#) third party product.
  - When the optional vbWatchdog third party product is installed, this module will display and log error messages for all unhandled exceptions as well as handled ones. Each error trap is also displayed in the VBA immediate window when in Development mode.
  - Without the vbWatchdog product installed, code must be inserted into each VBA Subroutine and Function in order to handle errors and call the Error Processing module. The optional [MZ-Tools](#) third party product helps with code insertion and line numbering in this case.
- **Source Code Versioning (optional):** The Framework supports optional source code versioning. It uses [OASIS](#) and [TortoiseSVN](#) to interface between Access and a repository. The repository may be a [Sub Version](#)(SVN) repository on a remote server, or a single-user repository on the developer's PC. In the latter case, the repository can be built and maintained by TortoiseSVN.
  - Easily see the detailed differences between two versions of a module.
  - See all the changes between two versions of an application or groups of selected modules.
  - Create a record of each release of your software.
  - When using a server-based repository you can have multiple developers sharing their work on a common application.

This setup is not required, and you may manually enter a Version String for display instead.

- **VBA Code Library:** Over 135 productivity enhancing functions are included for your use. These include the areas of
  - Control of the AJS infrastructure like GblErrorHandler() and putlog()
  - Retrieval of over 60 Customization Options with AJS.GetOpt()
  - File Utilities like file selection dialogs and IsValidFileName()
  - General Utilities like IsInArray(), TableExists() and EmptyTable()
  - Form Utilities like Change\_Height\_Of\_Subform() and Form Rescaling.
  - Windows operating system functions like WaitMsec() and Shell() with Wait or no-Wait options.
  - See the AJS Checklist and Reference documents for complete descriptions