WID API Project

Draft Statement of Work

# background

The Workforce Information Database (WID) is a database structure that standardizes how labor market information is stored. It was developed to enable data sharing between states and to facilitate the development of third-party applications that use state labor market information. The WID structure is defined and maintained by the Analyst Resource Center (ARC).

The WID structure defines database tables, fields, and relationships independent of any particular database platform (e.g. Oracle, SQL Server, MySQL, etc). However, any application that uses a particular WID database will very likely need to access it in a platform-specific way. This leads to a situation in which states have WID databases that are consistently defined, yet may still have difficulty sharing data between different platforms. Also, vendor applications are usually platform-specific, so the choice of platform may determine the solutions available, and vice versa.

One way to make data more widely available in a platform-neutral manner is to create a layer of web services. These services would provide WID data via http in a standard data format, such as XML or JSON. These web services could provide a foundation for labor market information web applications, or for integrating LMI into other applications.

Most application development now uses web services as the method of choice to access data from a database. This will make application development easier for states, and give them a workable application model that matches the database model.

# 2.0 purpose

The purpose of this project is to create an API specification for a set of web services for the WID. By working together with interested parties from multiple states, it is hoped that duplication of effort can be avoided, that design issues can be thought through carefully, and a common standard agreed upon.

Once an API specification has been developed, it can be published by the ARC and made available for any state (or other third party) to implement. The goal is for all implementations of the API to be as consistent as possible.

# 3.0 personnel

The WID API workgroup consists of the following ARC members and staff:

|  |  |  |
| --- | --- | --- |
| **Name** | **Email** | **State** |
| Steve Duthie | sduthie@mt.gov | Montana |
| Rajani Edupalli  | Rajani.edupalli@vec.virginia.gov | Virginia |
| Greg Garner | thomas.garner@vec.virginia.gov | Virginia |
| Joe Jaehnke | Joseph.Jaehnke@dwd.wisconsin.gov | Wisconsin |
| Bill McMahon | bill.mcmahon@state.mn.us | Minnesota |
| Amanda Rohrer | amanda.rohrer@state.mn.us | Minnesota |
| John Pearce | John.F.PEARCE@oregon.gov | Oregon |
| Gary Sincick | Gary.M.SINCICK@oregon.gov | Oregon |
| Will Burchard | william.g.burchard@oregon.gov | Oregon |

Oregon will take the lead on coordinating the work group. Additional staff from Oregon may also be involved in the development effort. Other roles and responsibilities have yet to be defined and assigned.

# 3.0 scope

The goal of this project is to develop an API specification only, not an API implementation. As such, the focus of the project is on the API interface and design only.

## 3.1 In scope

In terms of design, the following items will need to be considered by the WID API workgroup:

* The overall structure and organization of the API.
* The list of tables that will be accessible in the first version of the API.
* Naming conventions for paths, parameters, and data.
* The various functions and methods that the API will allow on these tables.
* How the API will relate to the WID structure.
* How the API and the WID structure will be kept in sync over time.
* Any special requirements for certain tables (e.g. Employer Database)
* What kind of authentication/security will be needed for the API.
* How returned data should be structured (data model)
* Formats of returned data (e.g. JSON, XML, etc)
* How large data sets should be handled (e.g. paging)
* What metadata should be returned
* What http response codes to use
* What (if any) validation should be done on query parameters
* How to handle unimplemented methods (in the case of partial or incomplete implementations)
* What links to related API calls should be returned
* If possible, the API should be designed to accommodate third-party tools, such as Tableau

## 3.2 out of scope

* An implementation of the API (i.e. an actual web service based on the API). Some implementation may occur during the development of the API specification in order to test certain features of the specification, but the deliverable for this phase of the project is just the specification, not a functioning web service.
* An API for inserting, updating, or deleting data from the database. This may be added in a future version, or addressed in a separate project.

# 4.0 Applicable standards

It is proposed that the WID API be defined using the Open API 3.0 specification (OAS). This is the current industry standard, and is supported by companies such as Google, Microsoft, Oracle, and IBM.

# 5.0 SOFTWARE REQUIREMENTS

The development of the API will require software for team communication and collaboration. Basecamp will be used for this purpose, and accounts can be provided by the State of Montana.

Also, software will be needed to create the actual WID API. This will be SwaggerHub, a cloud-based platform with tools for creating and publishing an API, as well as code generation tools which can be used to assist with API implementations.

# 6.0 DELIVERABLES

The following are the proposed deliverables for this first phase of the WID API project:

* A document that summarizes the major design decisions and philosophy of the API. This will help future development and extension of the API.
* A WID API specification in Open API 3.0 format. This can be made available for download from the ARC website. It can be used to generate HTML documentation and generate code for implementation.
* Supporting documentation that would help states or other third parties to understand the API and to implement it in a consistent fashion.

# 7.0 Milestones and schedule

|  |  |  |  |
| --- | --- | --- | --- |
| Milestone 1 | Project Definition | Define project scope, statement of work, and high-level design guidelines. | 1Q2019 |
| Milestone 2 | Project Requirements | Develop specific requirements for the WID API specification. | 2Q2019 |
| Milestone 3 | API Specification | Create the WID API specification in SwaggerHub. | 3Q2019 |
| Milestone 4 | API Publication | Publish API specification, documentation, and implementation guide on the ARC website. | 4Q2019 |