



SPARSH
(case study)

SPARSH

Company Profile

The revolutionary First Indian Language Portal www.webdunia.com was launched in the year 2000. Webdunia has played a very critical role in advancing language content in the country along with specializations in providing the Localization Services, Mobile VAS, Language Technology, Enterprise & Web Solutions.

Business Situation

A department of Central Government of India provides a wide range of services to general public. This department used to publish information related to its citizen centric services by means of an application developed in VB6 and displayed through kiosks on multiple locations. Their training centre decided to get this Application redeveloped using Windows Presentation Format and introduce it to its users in Hindi and other Indian Languages.

Solution

This application is a touch screen based application running at about 450 kiosks of the department. The application facilitates end users to know information about various services provided by the department in their own language.

Benefits

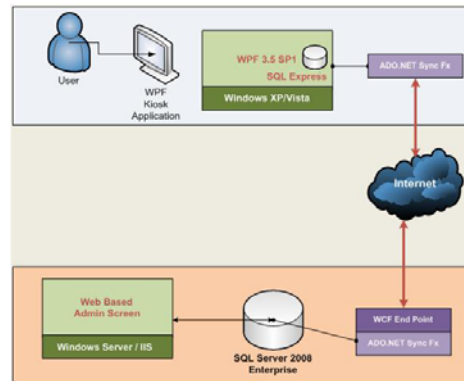
- User Friendly
- Attractive UI
- Multilingual support
- Workflow based data administration
- Centralized database server with each client having own set of database
- Auto-sync of data from central server
- Click Once Deployment
- Lingual keyboard to type in supported languages
- Content Localization

This is a touch screen based application running at about 450 kiosks of a department of Central Government of India. The application facilitates end users to know information about various services provided by the department in their own language.

Technology:

- **Client machines (Kiosks)**
 - WPF 3.5 SP1 Frontend
 - SQL Express 2008 as backend
 - ADO.NET Sync Framework for syncing with central server
- **Server:**
 - Admin application developed in ASP.NET 3.5
 - Data stored in SQL Server 2008 database
 - ADO.NET Sync Framework

Solution Architecture



Developed using latest Microsoft Technologies –

1. WPF, WCF (.NET Framework 3.5)
2. SQL server 2008/SQL Express 2008
3. ADO.NET Sync Framework



Current Situation

system at many cybercafés is manual. Hence cybercafé operators find it difficult to bill the customers properly for the time which they have spent in the cybercafé. Operators find it difficult to bill the customers for the time which they have spent in the cybercafé. By using the Billing and Metering Tool, you can accurately track and



Multilingual/Unicode Support

Application currently supports 14 languages -

- Assamese
- Bengali
- English
- Hindi
- Gujarati
- Kannada
- Konkani
- Malayalam
- Marathi
- Oriya
- Punjabi
- Tamil
- Telugu
- Urdu

Application Localization Approach

The application fetches localized data from:

- Database

We have language wise rows for a record and we select a row according to the current language.

- Resource files

Localized strings are kept in an XML file. For every language there is a separate XML file. When application starts it loads language specific xml file and creates a Dictionary of Items (key value pair). Using Markup Extension class.NET 3.0, we create a custom markup extension to bind the localized data in XAML itself.

Content Localization

Content Localization for this application consisted of total six different stages. The stages are as follows;

- Extraction of localizable strings.
- Analysis of the file to be localized
- UI Localization
- UA Localization
- Docs Localization
- LQA

Please find the brief details of the above stages below;

Extraction of the localizable strings

- In this stage we extract all the localizable UI strings from the files to be localized.

Analysis of the file to be localized

- All the raw files (UI, UA & Docs) to be translated are analyzed against a blank TM to get the approximate volume of the project using Cat tool available. The details about all the fuzzy matches are stored in the log report.
- Then Style Guide and Glossary is prepared by individual Language experts.
- Our Engineers then create a list of all terminologies (Product and service) names which has to be retained in English.
- Other specific guide lines are prepared such as translation of Windows navigation path in a specified format etc.
- The files to be localized are then categorized in to UI, UA and doc files and push forward for translation in a sequence.





...ts Situation
 ...ystem at many cybercafés is manual. Hence cybercafé
 ...ators find it difficult to bill the customers properly for
 ...ation which they have spent in the cybercafé.
 ...ators find it difficult to bill the customers for
 ...ation which they have spent in the cybercafé.
 ...e Billing and Metering Tool, you can accurately track and



UI Localization

- The file containing the UI terms are filtered for sorting out the duplicate/repeated strings.
- Then we get all the repeated UI strings translated from our native language linguist.
- The translations are stored in a database, mainly known as TM (Translation Memory).

Please Note: Translation Memory is provided by our linguist after completion of every stage to measure the accurate volume.

- Upon receiving the TM and the localized file, we auto translate them for localizing repeated terms. Once repeated terms are localized, remaining content's volume is measured once again.

Please Note: Auto translation is done using CAT tools.

UA Localization

- The content of UA files are extracted in .htm formats.
- The .htm files are then forwarded to the linguist team for translation along with the new log report.
- The deliverables we receive after localization of this stage are clean, unclean and TM file.
- Upon receiving all the localized .htm files, our Engineers compiled them to a chm (Compiled help manager).

Documents Localization

- In this stage the doc are localized separately with all guide lines in place.
- Once this is done we are through with the localization process of the Product. To ensure quality and consistency across different stages we then process the GUI/build screen shots for linguistic QA.

Language QA

- In this stage our linguists performs a thorough checks for linguistic quality of the localized build.
- The process includes taking screenshot from the localized build and sending them to the linguist along with the source screenshots to have a proper reference of the context of the translation and appearance of the localized strings during run time.
- The process is repeated twice to maintain consistency and regression of the fixed terms.
- In this stage all the cosmetic and non cosmetic issues are identified and fixed.

With this we complete the content localization of the product at our end.

