Vol. 10, No. 2, 2023, Pages 1-18

JIEMS
Journal of Industrial Engineering and Management Studies

journal homepage: www.jiems.icms.ac.ir

Software development and programming tools- a survey and recommendation for organization of small industries and industrial towns of Iran

Hassan Rashidi*[1], Zeynab Rashidi[2], Latifeh Pour Mohammad Bagher [1], Mohammad Bahrani[1]

[1] Department of Statistics, Mathematics and Computer Science, Allameh Tabataba'i University, Tehran, Iran.

[2] Department of Instructional Technology, Allameh Tabataba'i University, Tehran, Iran.

Received: Mar 2023-31/ Revised: Jun 2023-09/ Accepted: Sep 2023-11

Abstract

In today's world, software tools play an important role in speeding up software development, reducing development costs and human efforts, as well as increasing reliability. In software development by tools, choosing a suitable tool will be a difficult task because many of them are available with different capabilities. On the other hand, little research has focused on the classification of these tools and their comparison. This paper aims to perform a literature review of software development tools and to propose architectures for the requirement of the Organization of Small Industries and Industrial Towns of Iran (OSIITI), in Iran. We did a survey over more than 50 software development and programming tools. The results of this survey identified ten classes, namely (a) Database Tools; (b) Integrated Development Environment; (c) Software Development Frameworks; (d) Data Science Tools; (e) Source Control Tools; (f) DevOps Tools; (g) Unified modeling Language (UML) Tools; (h) Cloud Tools for Software Development; (h) Prototyping Tools; and (j) Notifications Programs. For each class, we collected the most software tools that are currently used with their major features. After that, two architectures, based on layered and service-oriented patterns are proposed for OSIITI. The ten specified classes, along with the tools in each class, are very useful for organizations like OSIITI who want to develop software, for both small and large projects.

Keywords: Software Development, Tools, Programming.

Paper Type: Original Research

1. Introduction

Nowadays, Computer Aided Software Engineering (CASE) design tools help developers to produce a system. They enable developers to graphically depict the logic of a program through flow diagrams. They also help programmers to easily design the user interface of the system, its layout as well as the reports it will generate. The user interface provides facilities for the interaction between users and computers. Moreover, it shows a graphical arrangement of input and/or output elements, as well as the location of the data and information on the screen or printout.

This study began with the challenges facing the Organization of Small Industries and Industrial Towns of Iran (OSIITI). This organization is responsible for supporting and developing small industries and studying and organizing the construction of industrial towns and industrial areas in Iran. The most important development missions of the organization in the current conditions are as follows:

(a) Facilitating, encouraging, and laying the foundation in order to create a suitable connection and organization between small, medium, and large industries;

(b) Policymaking and planning for the development of business clusters and networks and issuing the required licenses;

(c) Facilitating and encouraging the creation and development of R&D centers and innovation centers at the level of industrial fields;

(d) Managing the provision of required financial resources and investment and trying to provide collateral for small industries through investment guarantee funds;

* Corresponding Author: hrashi@gmail.com

(e) Setting the stage for issuing technical and engineering services in the field of creating industrial investment infrastructures (towns and industrial areas) and small industries;

(f) Planning for the empowerment of manpower and support of programs to provide specialized and skilled manpower needed by industries;

(g) Policy-making and planning in the field of drafting service descriptions, guidelines, and regulations and draft by-laws for the creation and development of towns and industrial areas and their supervision;

(h) Policy-making, planning, and drafting a strategic plan for the creation and development of technology towns, technology, and business service centers and information technology complexes, and software services;

(i) Policymaking, planning, and monitoring the creation of workshop complexes, service centers, and the creation of special economic zones under the responsibility.

According to the latest information from the deputy of OSIITI, the total number of stagnant units in the towns and industrial areas of Iran is seven thousand and 151 industrial units. The main challenges of this organization are that: (a) around 44% of these units have the problem of lack of liquidity, (b) 16% lack of demand and market, (c) 10% wear out machines, (d) 10% weak technology, (e) 11% lack of raw materials, (f) 9% have legal problems and (g) 8% have infrastructure problems. In fact, they have several challenges to provide infrastructure for these problems, mainly in: (i) selling products and supplying services to domestic and foreign markets, (ii) obtaining raw materials, and (iii) providing capital for development. To respond to the challenges above, this organization must seek technologies and platforms that are used to collect, store, analyze and access data as well as help organizational users to make better business decisions.

This paper is motivated to survey the software development tools and to provide platforms for OSIITI. The remaining sections of this paper are structured as follows. In Section 2, we review the latest research devoted to classification and studying tools for software development. In Section 3, we review the major database management software and software development tools. In this section, the major attributes of the latest tools are presented. In Section 4, we propose a couple of architectures for OSIITI. Section 5 is considered for the summary and conclusion.

2. Related Works

There is limited research devoted to classification and survey software and programming tools. In this section, the latest research on these tools is reviewed. The development of international software with global standards requires many supports of software tools with special capabilities. Portillo Rodríguez et al. (2010) explained some desirable capabilities to support these tools in the field of internationalization and also how these capabilities are related to the main challenges in the international areas. The authors conducted a survey on the capabilities of these tools. The tools included in the survey were classified using ISO/IEC 12207 standard processes to determine which software development processes are supported by each tool. The classification of tools according to their capabilities is also included, where the specific capabilities of each tool are shown. In recent years, mobile applications (apps) attracted many investments and interests. Apps are software applications or computer programs designed to execute on mobile devices such as tablets phones, or watches. Tavakoli et al. (2018) did a survey over the tools used for the development and intelligent mining procedures behind mobile applications. To gain a vision into the maturity of provision mining tools, this research also identified what challenges they are facing and found out what techniques are included in the tools for app development. The results of this survey provide useful knowledge for the development of intelligent apps with more effective mining techniques and tools. Fregnan et al. (2019) did a survey on software coupling relations and development tools. The first goal of this study was to present a classification of different types of couple relationships in software components, along with criteria for their measurement. The second one includes providing an overview of the tools that have been proposed so far by the academic society of software engineering to derive these criteria. This study conducted a systematic literature review in software engineering. Publicly available scientific research databases were used to retrieve cited publications. These resources were queried using the keywords inherent to the software coupling. They considered publications from the years 2002 to 2017 and heavily cited earlier publications. A snowball method was utilized to retrieve the most relevant content. The results of this research showed there are four groups of coupling relationships, including semantic, structural, logical, and dynamic in software components. Moreover, it showed there is a fifth group of coupling relationships included in approaches as an independent group and criteria that have been developed for particular environments. Regarding the criteria in tools, this research identified three trends, emerging in recent years: scalability, extensibility, and the use of visualization methods. The research also found out tools that extract criteria relevant to each coupling group. This research presents some directions for software coupling and some applications of coupling components to follow. Developing criteria for particular environments and code smell detection are presented for possible future research directions. Ozkaya and Erata (2020) did a survey over challenges faced by software specialists in their software modeling activities. They did a survey of over 80 software specialists from 18 countries who developed software for different industries. They focused on eight categories of challenges in software modeling: (a) management the software complexity, (b) extending software modeling, (c)

domain-specific modeling environments for software development, (d) developing formal models for software development, (e) analyzing software models, (f) separating of concerns in codes, (g) transforming software models, and (h) managing software models. The results showed separation of concerns is a minor challenge in the categories for specialists while analyzing models is the major challenge in the categories. Several various concrete challenges in different categories have been detected, including (i) working the software modeling with steep learning-curve, (ii) inconsistencies and updating the software tools when the software semantics is extended, (iii) evolving the domain-specific language tools with new requirements, (iv) defining the formal models for software semantics in terms of the translations in formal languages, (v) decomposing software models into separate perspective and analyzing the consistencies between different perspective in software models, (vi) the consistent model transformation of software and the model synchronizations, (vii) using model inspectors for formal and semi-formal analysis, and (viii) versioning software models. Tian (2021) did a survey over source code and software architecture, which are two tangled artifacts that embody the interdependent design decisions made at two levels of abstraction, low-level and high-level, for software development. The authors utilized a combined-method of an online survey with 87 respondents from 37 countries and an interview with eight experts, who gave their views on the respondents about the relationships between software architecture and source code. The results revealed that software specialists mainly discuss five features of relationships between software architecture and source code. Moreover, it shows a few software specialists have agreed on dedicated approaches and software tools in the literature for identifying and analyzing the relations between software architecture and source code, regardless of recognizing the importance of such information for improving a system's quality attributes, particularly reliability and maintainability. The opinions obtained from this research showed that efforts and costs are the major obstacles that prevent software specialists from identifying, analyzing, and using the relationships between software architecture and source code. Moreover, the results have empirically recognized five features of relationships between software architecture and source code stated in the literature from the viewpoint of software specialists. The research suggested that an organized framework to manage the features of relations should be developed with dedicated approaches and tools. Moreover, it suggested considering the costs and benefits of maintaining the relationships, in the framework.

3. Software Development Tools- A Classification and Survey

In this paper, we did a survey over more than 50 software development and programming tools. The results of this survey identified nine classes, as shown in Table 1. These classes are :(a) Database Tools; (b) Integrated Development Environment; (c) Software Development Frameworks; (d) Data Science Tools; (e) Source Control Tools; (f) DevOps Tools; (g) Unified Modeling Language (UML) Tools; (h) Cloud Tools for Software Development; (h) Prototyping Tools; and (j) Notifications Programs. For each class, the latest major tools found are in the right column of Table 1.

Table1. Categories of Software Development Tools

Row	Category	The Latest Major Tools found
1	Databases	MySQL, SQL **Server Management Studio**, Oracle **RDBMS**, **Salesforce**, Studio 3T, AQL Sentry, DB Schema
2	Integrated Development Environment (IDE)	Apache NetBeans, AWS Cloud9, Zend Studio, Rad Studio, Linx
3	Software Development Frameworks	Bootstrap, HTML5 Builder, Visual Online
4	Data Science Tools	Data Studio, Python, R
5	Source Control Tools	Gitub, BitBucket
6	DevOps Tools	Red Hat CodeReady Workspaces, Jenkins, Docker, Puppet, Apache Maven, Gradle, CircleCI, Bamboo, TeamCity, Travis CI, Buddy
7	UML Tools	Enterprise Architect, yUML
8	Cloud Tools for Software Development	Azure, Kwatee, Atom, CodeLobster, CodeCharge Studio
9	Prototyping Tools	Azure, Pencil, Mockitt, Adobe XD, Marvel, Moqups, Proto.io, Invision, Flinto, JustinMind,Origami
10	Notifications Tools/Programs	SendBird, Slack

3.1. Software Development- Database Tools

In this paper, we did a survey over many software development Environments for database management. The results of this survey identified seven software, as shown in Table 2. A short description along with the main features of these tools are explained short in the first and second columns of the table, respectively.

Table 2. The Current Software Development Environment for Database

Tool [Ref]	A Short Description	Main Features
MySQL [1]	It is an open-source relational database.	• It is used by topmost companies such as Google, Twitter, YouTube, LinkedIn, and PayPal. • It contains many popular web applications like WordPress. • It is a good choice to learn just one database because of its popularity and ease of use alone.
SQL Server Management Studio [2]	It is an all-in-one solution to profiling and tuning database performance.	• It provides access and management for SQL Server by a powerful set of user-friendly graphical tools and a variety of script editors. • It can be used by database administrators and developers with all levels of skills.
Oracle RDBMS [3]	It is an object-relational database management software and the most widely used.	• Its latest version incorporates cloud computing. • It is secure and supports UNIX, Linux, and multiple Windows versions. • It supports large databases and requires relatively less storage with moderate CPU time to process data.
Salesforce [4]	It is a useful database management tool that offers a diverse infrastructure of software products designed to help teams.	• Within this tool, the database structure is inherently completed for software developers. • The software solutions can be built based on workflow and business requirements.
Studio 3T [5]	It is used for MongoDB which helps developers to build queries fast, produce instant code, import/export in multiple formats, and much more.	• It supports Query MongoDB with IntelliShell, Visual Query Builder, and SQL Query tools. • It supports Data Masking tools that allow data compliance and boosts security with controlling field-level data complication. • It allows exporting to MongoDB from JSON, CSV, SQL, and BSON/Mongo-dump, • It provides previewing of output documents when developers make changes. • It supports migration from MongoDB to SQL and vice versa.
AQL Sentry [6]	It is a tool for database performance monitoring of the Data Platform, along with fast root cause analysis and visibility through Microsoft data estate.	• It helps developers to handle scheduled events and identifies resource contention in an Outlook-style calendar • It provides a proactive alerting and response system. • It allows developers to find and fix high-impact queries • It forecasts storage usages powered by predictive analytics • It helps developers to identify contributing problems in the OS and virtual environment.
DB Schema [7]	It is a visual database designer and manager tool for SQL, NoSQL, and Cloud databases. It is used by teams and deployed on multiple databases	• It has a capability for visual design & interaction with the database schema, designing schema by team developers and deploying it on multiple databases, • It generates migration scripts between different versions of the schema. • It supports HTML5 Documentation, Visual Query Builder, Relational Data Explorer, and Interactive Diagrams. • It supports Schema Synchronization, Data Loader, Random Data Generator, and Database Reports • It has a SQL Editor with auto-completion

3.2. Integrated Development Environment (IDE)

[1] Available at https://www.mysql.com

[2] Available at https://learn.microsoft.com/en-us/sql/ssms/

[3] Available at https://www.oracle.com

[4] Available at https://www.salesforce.com

[5] Available at https://studio3t.com

6 Available at https://www.solarwinds.com/sql-sentry

7 Available at https://shop.dbschema.com

The integrated development environments (IDEs) are software tools that help developers to make their program codes efficiently. They increase the productivity of developers by combining capabilities such as building, software editing, testing, and wrapping for easy-to-use applications. Table 3- summarizes the major current tools with the integrated development environment.

Table 3. The Current Tools with Integrated Development Environment (IDE)

Tool [Ref]	A Short Description	Main Features
Apache NetBeans [9]	It is a popular, Free, open-source IDE. It is an application development tool that allows developing desktop, web applications, and mobile apps.	• It supports fast and smart code editing • It has an easy and efficient Project Management process • It supports rapidly development of User Interface • It helps to develop a bug-free code • Its IDE provides superior support for C/C++ and PHP developers • It can be deployed on several OS which supports Java, from Windows to Mac OS X systems to Linux
AWS Cloud9 [10]	It is an online integrated software development environment. It supports many languages like C, C++, Ruby, PHP, Python, Perl, JavaScript, and Node.js.	• It allows cloning the entire development environment. • It supports a Built-In terminal for command-line wizard • It helps developers to debug and set breakpoints and inspect variables of any JS/Node.js app • It helps to drag any file or Terminal to create multiple split views, easily. • It supports developers to choose an extensive set of default Runners to execute apps, such as Python, Ruby, PHP/Apache
Zend Studio [11]	It is a next-generation PHP IDE designed to produce apps to boost developers' productivity. It supports developers to write code faster, and debug more easily. It supports scalability according to the DPI settings of the underlying operating system.	• It supports to develop code faster with performance improvement techniques in indexing, validation and searching for PHP code. • It offers to debug capabilities with Xdebug, Zend Debugger, and Z-Ray integration • It supports an extensive plugin presented by the large Eclipse eco-system • It supports several development tools, including Docker and Git Flow • It helps to deploy PHP applications on any server for Amazon Microsoft Azure and AWS
Rad Studio [12]	It is a Powerful IDE for Building Native Apps on Windows, macOS, Android, iOS, and Linux. It allows developer to design Desktop and Mobile App UIs with less efforts in coding. It supports writing once, compile everywhere.	• It has a comprehensive Codebase for many major platforms • It is connected to more than 20 databases natively with FireDAC's, a high-speed direct access • It supports controlling an Up-to-date user interface, Windows Store support, WinRT APIs, HighDPI-related features, and • It uses the HTTP and REST client libraries, available on all platforms, to invoke REST services or AWS and Azure components. • It has a powerful CData Enterprise Connectors to access more than 70 data sources.
Linx [13]	It is a low-code IDE and server. IT professionals use Linx to quickly create custom automated business processes, integrate applications, expose web services, and efficiently manage large workloads.	• It is an Easy-to-use tool with a drag-and-drop interface • It supports more than 100 pre-built services and functions for rapid development • It supports One-click deployment to any remote or local Linx Server directly from the IDE. • Its Input and outputs embrace nearly any SQL and NoSQL. databases, many file formats (binary and text) or REST and Simple Access Object Protocol (SOAP) Web services • It supports live debugging with stepwise logic. • It supports backend processes by a timer, directory events, or message queue or finds out web services, and calls APIs via HTTP requests

3.3. Software Development Frameworks

A framework consists of a large reusable component for designing a particular application domain. For instance, there are components in frameworks for building graphical editors, database-driven web applications, accounting systems, and many more modules. A framework often contains a significant code base that is reused. In other words, software developers can suppose a framework as uncompleted applications that are missing several lower-level modules. Developers can add modules to a specific framework to complement and specialize it for a specific application. Contrasting to software product lines, which can be established by a company for its individual use, frameworks tend to be public resources such as library programs and toolboxes. Libraries programs are lower-level software modules that developers can include in their own codes, while frameworks are typically modules and high-level architectures whose low-level details must be filled. Frameworks and toolboxes make developing

[9] Available at https://netbeans.apache.org/

[10] Available at https://aws.amazon.com/cloud9/

[11] Available at https://www.zend.com/en/products/studio

[12] Available at https://guru99.click/UVWtl

[13] Available at https://linx.software

software easier for non-experts. Experts must design and implement specialized codes and improve the quality of the code in the final product. Table 4 summarizes the major current software development frameworks.

Table 4. The Current Software Development Frameworks

Tool [Ref]	A Short Description	Main Features
Bootstrap0 [14]	It is a framework for software development with CSS, HTML, and JS. It is a programming tool with many in-builds components that developers can easily drag and drop to assemble responsive web pages.	• It contains several ready-made blocks of software code. • It helps to ensure consistency irrespective of who's working on a specific project. • It contains an extensive list of components. • It has several base stylings for most HTML Elements • It can be customized allowing for the specific need of a project.
HTML5 Builder [15]	It is a tool for building web and mobile apps using a single PHP codebase, CSS3, HTML5, and JavaScript. Moreover, it supports targeting multiple mobile operating systems and Web browsers.	• It supports rapid application development to create cross-platform Apps with flexible Cloud services. • It contains a single visual framework to increase the speed of development. • It supports a collaborative workflow, in which designers and developers can work. • It creates enterprise or ISV web and mobile apps. • It contains a geolocation component in HTML5 Builder to create location-based browser and mobile applications.
Visual Online0 [16]	It is a tool with a collection of services to plan, build and release software across a variety of platforms. Moreover, it supports organizations and software developers, allowing them to produce the perfect development environment.	• It tracks and manages all concepts in Kanban framework or scrum panels with agile processes. • It improves code quality and catches issues early • It builds, manages, secures, and shares software components • It automates and simplifies Azure deployments • It supports several tools for manual, automated, and performance testing. • It supports the centralized version control system with free private repositories.

3.4. Data Science Tools

According to National Science Foundation in 2019, the data revolution has made its scientific ways in academia. Data science techniques are becoming more widely and deeply adopted in academia, and new data science opportunities are being added to the list at a steady clip. Data scientists are at the forefront of processing raw data and extracting their knowledge. They need to find out how to access newly available data sources, to exploit the new generation of powerful computing resources, and to develop new methods that take advantage of all three. Data science is a field of study that is at the intersection of data analysis and programming. In the knowledge age, any significant amount of data can be analyzed to determine trends and patterns of behavior. Typically, data scientists do not develop software for markets. Instead, they develop small to large programs with the specific purpose of analyzing data. Although data itself could not look like the most fascinating topic, data science has many useful applications. Developers can use data science software to detect fraud in insurance companies, banking, and perform analysis of medical images, or even in marketing and advertising. Panagiotis et al. (2015) provided an overview of open-source tools for data science and proposed a taxonomy scheme, which can be used for studying open-source data science tools. The proposed classification scheme is based on four group features: (a) general characteristics, (b) operational characteristics, (c) data mining characteristics, and (d) project activities. This research then used the proposed scheme to review the seventy identified open-source software. The features of seventy open-source tools are documented based on the criteria of the four group features. Table 5 summarizes the major current data science tools.

[14] Available at https://getbootstrap.com/

[15] Available at https://apps.apple.com/us/app/html5-builder/id834931010

[16] Available at https://visualstudio.microsoft.com/

Table 5. The Current Data Science Tools

Tool [Ref]	A Short Description	Main Features
Data studio[17]	It is a collaborative data science software platform, which can be used by data analysts, data scientists, and engineers to build, prototype and explore, and deliver their data products.	It provides a facility to visualize data for stages of data analysis. It prepares, enriches, blends, and cleans data with more than eighty built-in functions. It bundles the whole workflow as a single deployable package for real-time predictions. It builds and optimizes several models in R or Python and integrates any external ML library over APIs code.
Python [18]	Python is an "interpreted language" (not a "compiled language"), so it may use more CPU time than a compiled language. However, it's still a good choice because of the time it saves developers (due to ease of use).	It is open-source with free installation. It attracts a great online community and is very easy to learn. The ability to become a common time for data science and web-based analytics products Of course, along with the advantages, Python also has some disadvantages, the most important of which are mentioned in Adame.
R[19]	R provides a software environment with programming languages for statistical computing and data science. It was designed and implemented based on S and Scheme languages.	It was released under the GNU General Public License by R Core Team and made available to the public for free. It is very similar to the S-PLUS statistical software in terms of commands. It covers most fields of applied statistics such as time series analysis, linear and non-linear regression, classical hypothesis testing, coding, clustering, etc It is a powerful software for creating graphics and charts.

3.5. Source Control Tools

Source code management tools are designed to manage evolving codebases of systems to enable the storage and retrieval of different versions of their components and entire software project. With source control tools, developers can work in parallel without interfering with each other and can integrate their own codes with the codes created by other developers. Source control or version control tools have several features for managing and tracking changes to program code. These tools provide a history and backups of code development and help resolve conflicts when integrating multi-source programs. Table 6 summarizes the major Current Source Control Tools.

Table 6. The Current Source Control Tools

Tool [Ref]	A Short Description	Main Features
GitHub0 [20]	It is a tool to help developers to build software, review code, and manage projects. It offers the right tool for different development jobs.	It can be integrated with GitHub's management tools easily, get done, and stay aligned It supports easy documentation along with quality coding It helps developers to host their documentations directly from repositories. It allows all program codes in a single place.
Bitbucket [21]	It is a version control tool that helps developers to easy collaboration among the software development team. Moreover, it can be integrated with JIRA, which is a project and issue-managing app.	It has a feature for permissions access to ensure that the only right people can make changes to the program code It supports organizing repositories into projects for development team to focus on a process, goal, and product by It helps to show build results from CI system It can be integrated with existing workflow to streamline software development process

3.6. DevOps Tools

Many companies still use some traditional models of development, deployment, and support. There are inevitable delays and overheads in these models. To speed up the development, support processes, and release software for

[17] Available at https://www.dataiku.com/

[18] Available at https://www.python.org

[19] R for Data Science, Available at https://r4ds.had.co.nz

[20] Available at https://github.com/

[21] Available at https://bitbucket.org/

engineers, an alternate approach called DevOps (Development + Operations) has emerged. Nowadays, more and more companies are using this approach with a single team responsible for all of the activities, depicted in Figure 1. These three activities can have several developments and widespread adoption of DevOps.

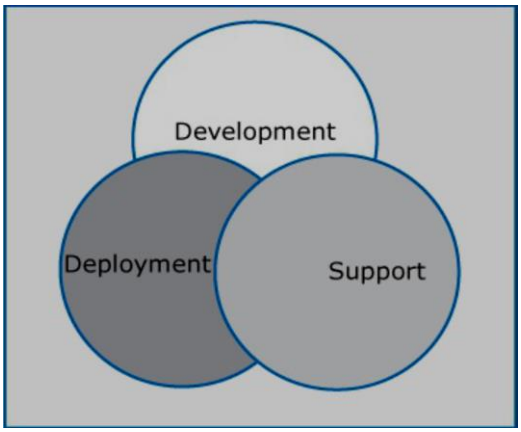

Figure 1: Multi-Skilled DevOps Team

DovOps is a software production process that is based on communication and cooperation as much as possible between software development teams and executive teams. During this process, software development operations as well as the implementation of infrastructure changes become automatic, and the overall goal of such a process is to create a culture. The aim is to produce, test, and release software in a fast, continuous, and reliable way. Table 7 summarizes the major current DevOps tools.

Table 7. The Current DevOps Tools

Tool [Ref]	A Short Description	Main Features
Red Hat Code Ready [22]	It is a tool to automate applications or micro-services for running on any number of servers. Moreover, it supports fully automates deployments of binary and text files from many target servers.	• It supports a wide range of operating systems including Windows, Linux, Solaris, Mac OS X, etc. • It manages environment-specific configuration parameters for applications. • It supports generating command-line installers for installation on any environment without installation and configuration files. • It reduces several operational risks in the software development process • Its friendly web interface allows configuring deployments efficiently and painlessly
Jenkins [23]	It is a free open-source automation server that supports automated software development processes such as building, Continuous Integration and Continuous Delivery (CI/CD), deployment, and testing. Being a Java-based DevOps tool, it supports developers to easily distribute systems across hardware to speed up development, testing, and deployment.	• It supports more than 100 plugins to integrate with all tools in the Continuous Integration and Continuous Delivery (CI/CD) toolchain, such as Amazon EC2, Git, Maven, etc. • It is a standalone program developed in Java and works on major platforms including Windows, Linux, Unix and macOS. • It allows developers to easily set up and configure Jenkins through its simple web interface along with monitoring capabilities and built-in debugging. • It is highly extensible through plugins, so that developers can add a wide range of functionality to it.
Docker [24]	It is a tool to help developers to build, package and then deploy code easily and quickly through containers with required dependencies, rather than virtual machines. It ensures that the software development environment is maintained uniformly throughout all stages of the DevOps cycle, from development to staging and production.	• It works easily with GCP (Google Cloud Platform) and AWS (Amazon Web Services). • It facilitates distributed development. • It allows adding features and making modifications easily • It runs on Windows, Linux, and macOS. • It can be integrated well with pipeline tools like GitHub and CircleCI. • It supports both commercial solutions and open source. • It is used by companies like Netflix, PayPal, Adobe, AT&T, and more. • It allows developers to produce Docker images and use them in the development environment by operations teams for testing and deployment.

[22] Available at https://developers.redhat.com/

[23] Available at https://www.jenkins.io

[24] Available at https://www.docker.com

Puppet [25]	It is an open-source tool for software configuration management. It helps to manage various stages of the software life cycle such as IT infrastructure provisioning, connectivity, configuration and management of software and operating system components in data centers and cloud infrastructures.	• It was developed in C++, Clojure, and Ruby, working with Windows, Linux, and Unix operating systems. • It is a model-driven tool that involves limited use of a programming language. • It uses its own programming language to make the system configuration. • It helps to reduce manual errors and supports development team. • It provides node management, orchestration, access control, reporting, and product support.
Apache Maven [26]	It is mainly used for Java-based projects and serves as a project management tool. Moreover, it is also used to develop and manage software projects written in Ruby, Scala, C #, etc.	• It manages the construction, documentation and reporting of a project. • It has predefined targets for performing tasks such as packaging and code compilation. • It supports Java libraries and Maven plugins from its repositories to speed up the development process.

Table 7. (Continued)- The Current DevOps Tools

Tool [Ref]	A Short Description	Main Features
Gradle [27]	It is an open-source DevOps tool for building automation, particularly for multilingual applications or software development. It was developed in Java, Groovy, and Kotlin. It is used to automate software development, testing and deployment to improve speed.	• It has a complete API, along with a rich ecosystem of plugins and integrations to help developers automate end-to-end software delivery, development, and integration. • It introduces a Groovy-based DSL and Kotlin-based. • It supports acyclic and directed graphs to prioritize tasks and execute through management. • It supports to development of mobile applications for micro-services. • It is a multipurpose tool that can be used by startups and companies.
CircleCI [28]	This CI/CD tool is based on the CircleCI cloud and offers good reliability and speed for the software development process. CircleCI automatically runs CI/CD pipelines in a virtual machine or container for easy testing. It notifies the development team immediately if a pipeline breaks down, and developers can also automate notifications using Slack.	• It automatically deploys code to a different environment to speed up the time to market. • It runs on Windows, macOS, Linux and supports Docker. • It supports many programming languages such as JavaScript, C++, PHP, Python, .NET, Ruby, etc. along with frameworks and toolchains. • It offers powerful storage options such as client cache, source code, and images to speed up pipeline processing and achieve optimal performance. • It allows developers to find and solve problems using SSH access. • It provides security with full virtual machine isolation, LDAP user management. • It offers a powerful dashboard to track the status and optimize pipeline processing.
Bamboo [29]	It is a tool to help developers to create build plans in multiple steps and trigger triggers. It is available as free and paid software, both. It is free for developing open-source software projects, while commercial organizations are charged based on the required build factors.	• It supports multiple builds. • It contains an intuitive and user-friendly interface. • It has tips, tools, and autocomplete. • Compared with Jenkins, its automation requires less configuration time with pre-built capabilities. • It can be integrated with many build tools and code repositories such as Mercurial, Git, Bitbucket, JIRA, Fisheye, Crucible, etc.
TeamCity [30]	It is a general-purpose CI/CD DevOps tool that offers more flexibility for different types of programming languages and workflow rules. It is used to build, deploy, and test various applications, containers, and packages.	• It has gated (input control) and promises to prevent developers from breaking source code in version control systems. • It is integrated for inspections, code coverage, and duplicate search. • It allows multiple tests and builds to run simultaneously on different platforms and environments. • It supports Ruby, Java, and .NET platforms. • It supports Mercurial, Subversion, and Git. • It can be integrated with IDEs like Eclipse, Visual Studio, and IntelliJ IDEA.
Buddy [31]	It is a tool to help seamless integration and deployment, along with feedback. It also supports developers to use Docker containers in addition to pre-installed frameworks and programming languages for building software.	• It is suitable for projects that use GitHub and Bitbucket code along with a simple and user-friendly interface • It enables faster deployment with change detection, parallelization, advanced caching, 360-degree optimization, RAM and virtual processor scaling, etc. • It supports developers to perform more than 100 actions like running and orchestrating Kubernetes clusters, SSH commands, Docker images, using Red Hat Package Manager (RPM) tools.

[25] Available at https://puppet.com
[26] Available at https://maven.apache.org
[27] Available at https://gradle.org
[28] Available at https://circleci.com
[29] Available at https://bamboogroup.eu
[30] Available at https://www.jetbrains.com/teamcity
[31] Available at https://buddy.works/

3.7. UML Tools

Unified Modeling Language (UML) is a language consisting of a standard set of symbols and notations for representing a software model. This language has a general syntax for creating a logical model of a system and is commonly used to describe a system that is understood at various viewpoints during analysis and design. Its syntax is designed to be independent of any specific target language, software process, or tool, but its generality has enough flexibility so that it can be customized using user-defined extensions to meet practically any language, tool, or process needs to fulfill. Although the syntax itself is well-defined and relatively easy to understand, it is much easier to apply to a specific project. To do this, it requires defining a set of semantics that is appropriate for a particular software architecture and process. We did a survey of well-known software for using UML. The results of this survey are summarized in Table 8.

Table 8. The Current UML Tools

Tool [Ref]	A Short Description	Main Features
Enterprise Architect[32]	It is a tool for requirement management that can be integrated seamlessly with other development tools. It creates model from the requirements.	• It improves business modeling and manages complex Data Effectively • It supports to build robust and maintainable systems • It can load extremely large models • It supports collaboration effectively globally and offers complete traceability. • It helps to generate documents in HTML. • It supports code execution to visual diagrams
yUML [33]	It is a tool to create diagrams based on text and can easily create them from Continuous Integration (CI) scripts or code. Moreover, it can simply manage an image programmatically using a GET or PUT. It is used by tool vendors to integrate yUML with wikis, blogs, and scripts.	• It supports chaining multiple items like this: [A]->[B]->[C] • It handles international characters including Korean and Chinese • It is more flexible use of hashes and commas in classes, like this [Customer|#count|login(user, password)] • If the user registers with the software, he/she can embed his/her named diagrams in Wikis. • Stereotypes can be done like this: [<<IEntity>>; Customer] • It supports developers to redefine a class with functions and attributes and the most detailed one will be used

3.8. Cloud Tools for Software Development

The high-speed networks and powerful multi-core computer hardware have led to the development of the "cloud". In simple terms, the cloud is a large number of remote server computers that are obtainable for rent by many companies that own these servers. Developers can rent as many server computers as they need, deploy their software on these servers and make them accessible to their clients. The clients can access these server computers through their personal computers or other mobile devices such as phones, tablets or laptops. Developers may rent a server computer and install their own software, or developers may pay for access to software products and services that are accessible in the cloud. Remote server computers are "virtual servers", meaning they are deployed on software rather than hardware. Certainly, many virtual servers can run concurrently on any cloud nodes using virtualization supports built into the hardware. Running many servers has very little impact and overhead on server performance. Running multiple virtual servers at the same time are certainly possible due to available powerful hardware. Table 9 summarizes the major current cloud tools for software development.

[32] Available at https://sparxsystems.com/products/ea/

[33] Available at https://yuml.me/

Table 9. The Current Cloud Tools for Software Development

Tool [Ref]	A Short Description	Main Features
Atom [34]	It is fully free and open source with a solid all-around text-editor. It can be customized to perform anything without a need to modify the configuration file.	• It runs across many popular operating systems such as Windows, OS X, or Linux • It simply browses and opens a single project or multiple projects in one window. • It supports splitting the Atom interface into multiple windows to compare and edit program codes across files. • It allows previewing, searching and replacing text types in a file or across the whole project. • It supports developers to develop code faster with flexibility, smart, and autocomplete
Codelobster [35]	It is a tool for simplification and streamlines PHP software development. process. Moreover, it supports CMS like Drupal, WordPress, Magento, and Joomla.	• It allows filling tags, attributes for a tag, and closing tags, automatically. • It includes HTML, CSS, PHP, JavaScript codes highlighting • It allows inspecting simply to find HTML elements and their styles buried deep in the page • It allows completing style property names and values, automatically. • It supports completing keywords, DOM elements, and their properties, automatically. • It supports PHP Advanced autocomplete
CodeCharge Studio [36]	It is a tool to develop applications rapidly and helps to develop Intranet applications, data-driven Web sites and enterprise Internet.	• It supports converting any database into a web application, rapidly. • It avoids costly errors and misspellings by producing consistent, well-structured code • It eliminates the efforts and time of programming tasks and creates scalable, robust Web Applications. • It helps to analyze and modify the generated codes to learn web technologies and it can be used to do projects in any environment
Azure0 [37]	It is a tool that extensively used by developers to build, test and deploy web applications.	• It supports an extensive range of frameworks, programming languages, operating systems, and devices • It supports building apps easily and quickly • It helps to detect threats easily and mitigates them. • It relies on the most trusted cloud. • It supports managing apps proactively and releasing mobile apps seamlessly
Kwatee0 [38]	It is a tool for Agile Deployment and automate applications or micro-services to any number of servers. It helps to fully automate deployments of binary and text files from any number of target servers.	• It supports to manage environment-specific configuration parameters for applications • It supports to generate command-line installers to be used for deployments in environments • It helps to eliminate the need for installation & configuration. • It helps to take out many operational risks in the software development process. • It has a friendly web interface that allows configuring deployments efficiently. • It supports several operating systems, including Windows, Linux, Solaris, Mac OS X, etc.
Travis CI [39]	It is a suitable tool for developing open-source projects and is an efficient continuous integration tool for developing and testing applications hosted on Bitbucket and GitHub.	• This cloud hosting service tool detects all new code that is automatically executed and the number of codes that are sent to the developers GitHub repository to help them build and test their software. • It allows quick setup and compatible with Mac, Linux, and iOS • It supports Python, Java, PHP, Perl. • It has an automatic deployment feature. • It provides new VMs for each build. • It supports an enterprise-level access control to improve security. • It has a feature for on-demand scaling capacity.

3.9. Prototyping Tools

Some software tools enable system developers to work with users to develop a prototype of a system. A prototype system is a model of a full-scale system under development and prototyping is the process of creating a system prototype, or model, to enable users to evaluate a preview of the system. A scalable system is a system whose design can handle an infinite expansion in the size of the database or the number of users. It is standard today for developers to build a scalable prototype system during development. A typical prototype system does not have full system functionality, but it does handle the main transaction-oriented procedures, produce common reports, and permit typical database queries. Prototyping emerged as a common approach in systems development out

[34] Available at https://atom.io/

[35] Available at https://www.codelobster.com

[36] Available at http://www.codecharge.com/index2.php

[37] Available at https://guru99.live/SLhzeZ

[38] Available at https://github.com/kwatee/agiledeployment

[39] Available at https://www.travis-ci.com

ofa need to give users a chance to review the proposed system while there is still time to make modifications to it based on user feedback. Previous systems development processes prevented users from seeing and using the system until after it was too late to make changes. A system specifications document is written before the system is fully designed. It is a documented description of an entire proposed system, from the functionality to the format of the output screens and reports; also known as specs (shorthand). Another document is functional specifications, which is a documented description of the system's logic from the user's perspective. There are three objectives in making a software prototype: (a) analyzing the current circumstances, (b) identifying information needs, and (c) developing a model of the proposed software. A target software is a fully functional system that is developed. Prototypes allow users to work with the functional aspects of the proposed system before the system is implemented. This hands-on experience helps users relate more precise information processing needs to the functional team. Table 10 shows the current prototyping tools with a short description and major features.

Table 10. The Current Prototyping Tools

Class [Ref]	A Short Description	Main Features
Axure[40]	It is capable to produce wireframes, and prototypes, and to make documentation. It is widely used by product managers, IT specialists, and business analysts, and around the world	• It helps to generate prototypes in HTML and provides a link for sharing. • It supports multiple developers to work on the same file at simultaneously. • It can be deployed on Microsoft IIS with MySQL or Microsoft SQL Server database. • It helps developers to produce and maintain widget libraries.
Pencil[41]	The pencil project is an open-source project for software prototype design	• It can be installed on all popular operating systems in the market • It can be installed as Firefox Extension • It provides an exclusive collection of components for iPhone and Android. • It helps to draw diagrams. • It supports various exports such as PDF, PNG, SVG files.
Adobe XD[42]	It lets developers reuse and recreate vectors to create wireframes, layouts, prototypes, and ready-made components using a single drawing program.	• It helps designers to work more efficiently by simply importing files from Adobe tools without any hassle. • It helps developers to consider integration with multiple products, such as projects in Photoshop and After Effects. • It helps developers to comment on their prototypes and see designs live on real devices when they share data directly.
Mockitt[43]	It is a collaborative tool for prototyping and team working that helps developers to visualize their ideas and enhance their customer experience.	• It allows developers to customize workflows. • It supports developers to design prototypes faster using built-in templates. • It can be used to customize project visibility. • It helps to organize communications effortlessly. • It offers a variety of browser-based inspection and export facilities. • It helps developers to create and reuse their own libraries.
Moqups[44]	It is an easy-to-use prototyping tool for developers to collaborate in real-time on mockups, wireframes, diagrams, and more. It provides a native support for drag-and-drop and Copy/Paste actions to import images from desktop or other applications.	• It includes many features to draw diagrams and to make decision trees, flowcharts, sitemaps, and more. • It provides a built-in library with more than one thousand icons from the most popular icon sets. • It helps to drag elements from the library of widgets and draw smart shapes with easy configuration. • It helps to turn the developer's wireframes or final designs into live prototypes by adding hotspots and interactions.
Marvel[45]	It is a browser-based prototyping tool and app that lets developers to upload their own image files and helps them add animations and transitions.	• It helps developers create real examples without code. • It supports developers to validate the idea through interactions with user testing. • It provides fast URL sharing and embed codes for mockups. • It allows direct upload of certain image types such as GIF, JPG, and PSD

[40] Available at https://www.axure.com/

[41] Available at https://pencil.evolus.vn/

[42] Available at https://xd.adobe.com/

[43] Available at https://mockitt.wondershare.com/download.html

[44] Available at https://www.moqups.com

[45] Available at https://www.marvel.com

Invision [46]	It is a general prototyping tool, used for project management. It helps developers to put design components in a proper workflow and state.	• It supports developers to upload multiple file types including JPG, PNG, GIF, PSD files. • It supports a stylish look with a flexible layer and a fully pixelated design. • It supports push and pull integration with apps like Dropbox, JIRA, and Trello.
JustinMind [47]	It is a prototypical tool for creating high-quality work with moderated cost. Developers can use it on their computers to do offline work anywhere.	• It allows developers to convert their prototypes into fully functional HTML documents and to prepare them easily available for viewing in any web browser. • It creates wireframes for web apps and websites that are compatible with multiple screen resolutions, for desktop and mobile applications. • It helps to access User Interface libraries and loads many extras.
Flinto[48]	It is used for designers to create simple and complex interactive prototypes.	• It supports web apps and iOS apps. • It supports developers to control all the different layers and complexities. • It uses a drag-and-drop component for prototyping. • It helps developers to add customizable navigation to each group in their app design.
Proto.io [49]	A prototypical tool for creating prototypes that allows developers to develop their ideas through a completely idealized design.	• It offers many opportunities for projects to create detailed and custom animations. • It allows developers to record user tests. • It provides shareable prototypes. • It supports VR prototyping.
Origami [50]	It was developed by Facebook to help teams for building and designing products. This tool allows developers to easily import their designs from Photoshop and Sketch into the program.	• It allows developers to call their prototype components with just one click. It supports developers to add a custom background to their design and view it in full screen. • It supports simulating using different devices. • In this tool, developer's clients or design team do not permit to directly comment on the project or version history.

[46] Available at https://www.invisionapp.com

[47] Available at https://www.justinmind.com/

[48] Available at https://www.flinto.com/

[49] Available at https://proto.io/

[50] Available at https://www.origamiway.com/easy-origami.shtml

3.10. Notifications Tools/Programs

These days, many businesses are trying to provide telecommuting infrastructure for their employees so that, in addition to removing the compulsion of employees to be at work, they can avoid additional costs as much as possible, and at the same time, work productivity and project progress. A notification is a message that displays an outside user interface to provide the developers with reminders, to make communication with other people, and to have other timely information from the apps. Developers can knock the notification to open their app or perform an action directly from the notification. We did a survey of well-known solutions in this field that can improve the productivity of affairs to a great extent in the software development environment. The results of this survey are summarized in Table 11.

Table 11. The Current Notifications Tools/Programs

Class [Ref]	A Short Description	Main Features
SendBird [51]	It is used as a messaging in Websites and Chat API for Mobile Apps. It provides scalability for a massive audience. It also prevents spam flooding in chat rooms.	• It reads and tracks the status of the messages sent to users. • It encompasses a soft bot to assist with customer support and provides recommendations for producing software. • It deals with push notifications and callbacks. • It reads the receipt and delivery status. • It helps to split or merge chat rooms on the audience volume automatically and offers continued engagement.
Slack [52]	Slack is simply a chat room for the entire company and its employees. Using Slack, a work team is divided into several cooking channels so that group members can discuss work together and develop project matters. By using Slack, the use of phone conversations or email during work will be greatly reduced	• It has comprehensive and practical facilities for team and group activities • It has the possibility of managing project actions and carrying them out by different members of the group • It has some features for reducing phone calls and email correspondence among members • It provides a safe platform for team members to talk and exchange opinions • It has a simple and user-friendly graphic environment
Collaborator [53]	It is a peer code and document review tool for supporting developer teams that take quality seriously with widespread review capabilities.	• It supports several version control tools including SVN, Git, CVS, TFS, Perforce, ClearCase, and RTC. • It is used for reviewing source code with documentations in Microsoft Word, Excel, PowerPoint, Adobe PDF, Visio Diagrams, Images, and Simulink models all in one tool. • It helps to enhance the code review process for Bit Bucket, GitLab, GitHub, and Azure DevOps • It provides a proof of review with electronic signatures and detailed documents to meet auditing and regulatory agreement standards (DO-178C, ISO26262, AS9100, 21 CFR Part 11, OWASP).

4. Proposed Architectures for OSIITI

In this section, a couple of architecture for Organization of Small Industries and Industrial Towns of Iran, are proposed. The first architecture is based on layered pattern and the second one is based on service-oriented. The details of these architectures along with the challenges are described below.

4.1. Proposed Architecture Based on Layered Pattern

Figure 2 shows a proposed layered architecture where each layer in the system consists of a number of logically related components. This formal layer is commonly used to represent how a system can be broken down into components, and each component provides substantial functionality to the system. Major web-based systems and mobile apps work based on event triggering. An event in the user interface, for example a mouse click, initiates actions to execute the user's selection. Furthermore, in the layered system, the flow of control is from top to bottom. User events at higher layers initiate actions at that layer, which in turn trigger events at lower layers. In contrast, many information is flowed in bottom-up direction in the system. Information is created at lower layers, then transformed at intermediate layers, and finally supplied to users at higher levels.

[51] Available at https://sendbird.com/

[52] Available at https://www.yasdl.com/

[53] Available at http://www.codecharge.com/index2.php

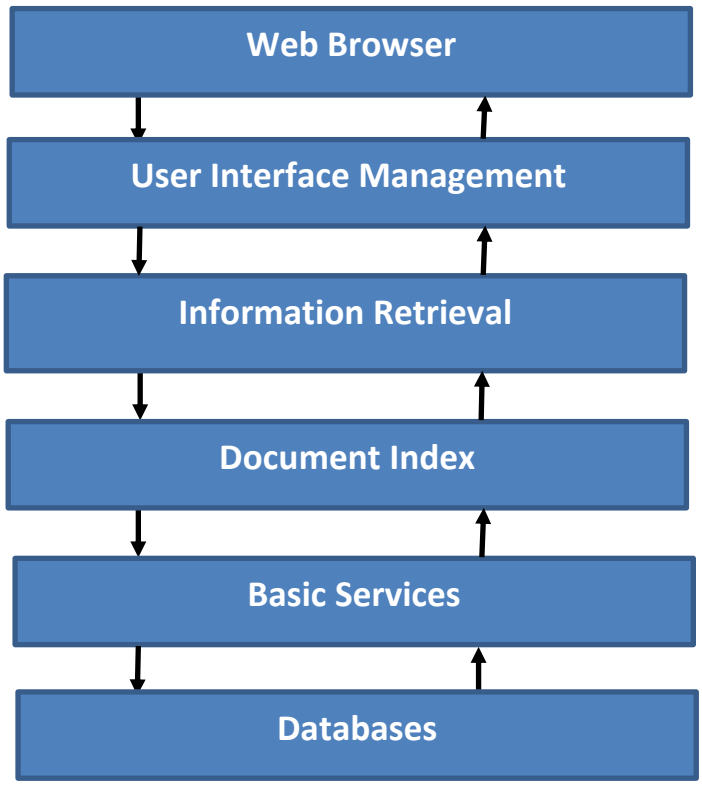

Figure 2. A Proposed Architecture based on Layered pattern for Organization of Small Industries and Industrial Towns of Iran (OSIITI).

Table 12 describes the functionalities of each layer in the proposed architecture. The databases include several databases required by OSIITI, such as selling products and supplying services to domestic (DB1) and foreign markets (DB2), obtaining raw materials (DB3), and providing capital for development (DB4).

Table 12. Functionalities of the Proposed Architecture, shown in Figure 2

Layer	Major Functions
Web Browser	Inputs Entry, Local Input Validation, User Interaction, Local Printing
User Interface Management	User Authentication and Authorization, Web Page Generation, Form and Query Management
Information Retrieval	Retrieval of Documents, Right Management, Payment and Accounting, Search
Document Index	Creation of Index files, Index Management, Index Querying
Basic Services	Performing Database Query, Query Validation, User Account Management and Logging
Databases	OSIITI databases including DB1, DB2, DB3, DB4.

4.2. Proposed Architecture Based on Service-Oriented Pattern

Figure 3 shows the service-oriented architecture for the requirement of Organization of Small Industries and Industrial Towns of Iran. Services in this architecture are components without a state, meaning they can be replicated and migrated from one computer to another. Moreover, many servers can provide several services. Usually, the service-oriented architecture is more scalable as demand increases and is more resilient to failure. Table 12 describes the technology, design decisions, and comments on each decision for this architecture.

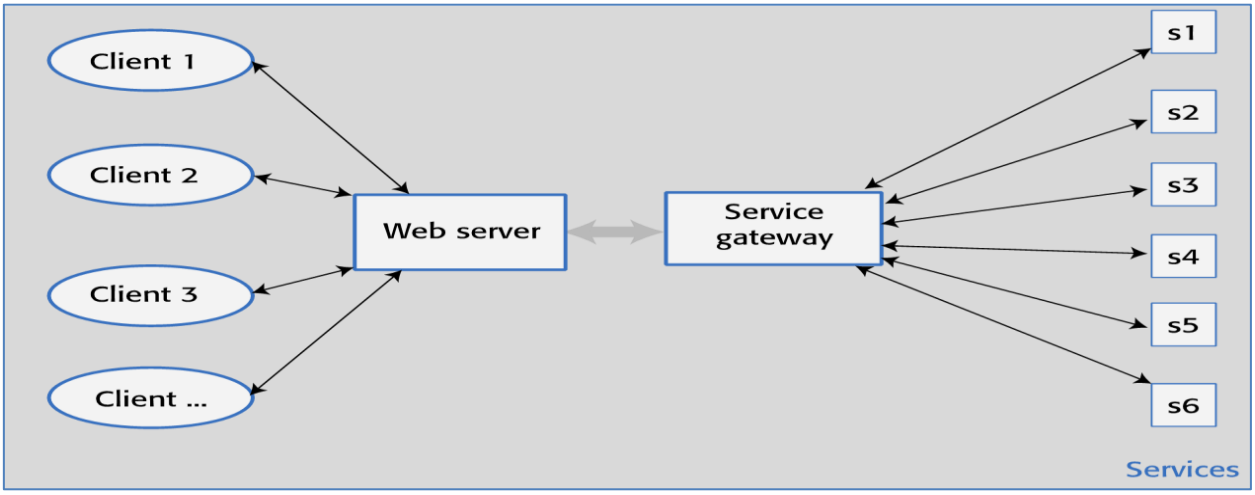

Figure 3. A Proposed Architecture for OSIITI, based on Service-Oriented Architecture

In Figure 3, the service gateway is a single access point and functions as a proxy for multiple services. This gateway supports routing, transformations, and ordinary processing across all the services. In fact, the service gateway is a single module that manages the service requests from multiple clients and the services provided by the servers. There are four steps that are usual of any service gateway:

- **Ordinary Processing**: When a message is received by the gateway, ordinary processing follows for all messages, such as creating and adding protocol-level headers or logging the message.
- **Service Identifying**: Any message that is processed by the gateway, it must be recognized and its service type must be specified. For instance, the message is queried to decide whether it is a request for the service provider A, B, or C.
- **Endpoint Routing**: When a message is determined to be sent to a particular service provider, it is mapped to an addressable endpoint on the network so that the message can be sent to the service provider.
- **Specific-Service Processing**: Any processing required for the specific target service is performed.

There are three challenges in this architectural choice. These challenges are related to data, the frequency of changing components, and execution platforms. Some useful guidelines are provided in the following, to respond these challenges.

- **Data Types and Data Updates**: If OSIITI wants to use structured data that can be updated by different system capabilities, it is usually better to have a single shared database that provides transaction management and locking. If data is distributed among services, OSIITI needs an approach to control consistency. This approach adds overhead to the system.
- **Frequency of Change:** If OSIITI anticipates that system components will be modified or replaced regularly, then separating these components as separate services will simplify those modifications.
- **System Execution Platform**: If OSIITI plans to deploy its systems in the cloud with accessing its users via the Internet, a service-oriented architecture is usually better to implement because this architecture supports scalability easily. If OSIITI wants to develop some specific products as business systems that run on local servers, a multi-tier architecture may be more applicable.

We suggest a couple of tool options to be used by OSIITI, including Open-Source tools and other development tool. The key question, comments, and recommendations for using these tools are explained in Table 13.

Table 13. Tool Options, Design Decisions and Comments on OSIITI's Decisions

Tool Options	Key Question	Comments and Recommendations
Open-source Tools	Are there any suitable open-source components that OSIITI could incorporate into its products?	• If OSIITI uses open-source software, its advantages are that OSIITI can reuse rather than develop new software tools. It will reduce development time and costs. • The disadvantages of using open-source tools for OSIITI is that its systems will be constrained by those tools without control over their evolution. • For OSIITI, the decision on the use of open-source tools also depends on the maturity, availability, and continuing support of open-source components. • Challenges related to licenses to use open-source tools impose some restrictions on how OSIITI can use the tools. A choice of open-source tools should depend on the type of products that OSIITI wants to develop, and the expertise of the development team.

Other Development tools	Does OSIITI use development tools embed software architecture assumptions that limit architectural choices?	• Software architecture is influenced by the development technologies, such as web application frameworks, and mobile development toolkits. These technologies consider some built-in assumptions and architectural patterns. OSIITI should be informed about and conform to these assumptions and patterns. • The development technology used by OSIITI may have an indirect effect on the architecture. Developers generally use architectural options with use common patterns. If OSIITI has enough experience with relational databases, for example, they might argue for this instead of a NoSQL database.

Table 14 shows the infrastructures of technology, including Database, Platforms, and Server, from, which OSIITI must exploit. This table also described the key question and design decisions, comments as well as recommendations for OSIITI.

Table 14. Technology, Design Decisions and Comments on OSIITI's Decisions

Technology	Key Question and Design Decisions	Comments and Recommendations
Server	Does OSIITI want to design the system to run on a public cloud or use dedicated in-house servers? for a public cloud, OSIITI should use an external option	• A key decision is that OSIITI wants to develop its systems to deploy on its own servers or to run on the cloud. • For systems that OSIITI wants to use as mobile apps, it almost reasonably makes sense to develop for the cloud. • For the systems that OSIITI wants to use as business products, it is a more difficult decision. Some businesses are concerned about cloud security and prefer to run their systems on in-house servers. • If OSIITI can predict a pattern of system usage, there is usually less need to design the systems to cope with large changes in demand. • An important decision for OSIITI, when its systems should be run on the cloud, is which cloud provider to choose.
Database	Does OSIITI want to use a relational SQL or an unstructured NoSQL database? There are two kinds of database: Relational databases and NoSQL databases.	• The data in Relational Databases, such as MySQL, is organized into structured tables. These Databases are particularly suitable for when OSIITI can predict changes in the data structures and situations where OSIITI requires transaction management. • In NoSQL databases, such as MongoDB, data is more flexible and user-defined. For data analysis, these databases are more flexible and potentially more effective than relational databases. They support organizing data hierarchically instead of flat tables, allowing for more effective and concurrent processing of "big data".
Platform	Does OSIITI want to deliver its products on a mobile app and/or a web platform? Its systems delivery can be as web-based or a mobile product or both	• In using mobile products by OSIITI, there are three issues: (a)Intermittent connectivity, (b) Power management; and (c) on-screen keyboards. • It must provide limited services without network connectivity. • Mobile devices don't have powerful CPU, so OSIITI must minimize its computationally intensive processing. • Battery life in Mobile is limited, so OSIITI must minimize the power consumption of the application. • Since on-screen keyboards are errors prone and slow, OSIITI must minimize input using the screen keyboard. To deal with these features, OSIITI typically requires separate browser-based and mobile-based versions of the front-end product. • To improve the performance and other features of OSIITI system, it requires a completely different decomposing architecture in different versions of platforms.

5. Summary and Conclusion

This research focused on software tools that help automation and software development processes. The purpose of these tools is to reduce human efforts, speed up software development, and increase reliability. These tools have a wide range and can be varied from single-purpose tools to the multipurpose tool. This paper made a literature review over software development tools and proposed architectures for the requirement of Organization of Small Industries and Industrial Towns of Iran (OSIITI), in Iran. In the survey, we studied more than 50 software development and programming tools. The results of this survey identified ten classes, namely (a) Database Tools; (b) Integrated Development Environment; (c) Software Development Frameworks; (d) Data Science Tools; (e) Source Control Tools; (f) DevOps Tools; (g) Unified Modeling Language (UML) Tools; (h) Cloud Tools for Software Development; (h) Prototyping Tools; and (j) Notifications Programs. For each class, several software tools were provided with their major features. A couple of architectures, based on layered and service-oriented patterns were proposed for OSIITI. These classes and tools are very helpful for organizations that want to develop software, in both small and large-scale projects.

References

Chen, S. (2023). A counterinsurgent (COIN) framework to defend against consumer activists. Journal of Brand Management, 30(4), 275-301.

Larsson, C. (2022). Cognitive Load in Advanced Systems: Registration of Swedish grades in the University Admissions process.

De Carlo, G., Langer, P., & Bork, D. (2022, October). Advanced visualization and interaction in GLSP-based web modeling: realizing semantic zoom and off-screen elements. In Proceedings of the 25th International Conference on Model Driven Engineering Languages and Systems (pp. 221-231).

Chakraborty, S., & Aithal, P. S. (2022). A Practical Approach to GIT Using Bitbucket, GitHub and SourceTree. International Journal of Applied Engineering and Management Letters (IJAEML), 6(2), 254-263.

Sommerville, I. (2020). Engineering software products (Vol. 355). London, UK: Pearson.

Barlas, P., Lanning, I., & Heavey, C. (2015). A survey of open-source data science tools. International Journal of Intelligent Computing and Cybernetics, 8(3), 232-261.

Portillo-Rodriguez, J., Vizcaino, A., Ebert, C., & Piattini, M. (2010, August). Tools to support global software development processes: a survey. In 2010 5th IEEE International Conference on Global Software Engineering (pp. 13-22). IEEE.

Tian, F., Liang, P., & Babar, M. A. (2022). Relationships between software architecture and source code in practice: An exploratory survey and interview. Information and Software Technology, 141, 106705.

Ozkaya, M., & Erata, F. (2020). Understanding practitioners' challenges on software modeling: A survey. Journal of Computer Languages, 58, 100963.

Fregnan, E., Baum, T., Palomba, F., & Bacchelli, A. (2019). A survey on software coupling relations and tools. Information and Software Technology, 107, 159-178.

Tavakoli, M., Zhao, L., Heydari, A., & Nenadić, G. (2018). Extracting useful software development information from mobile application reviews: A survey of intelligent mining techniques and tools. Expert Systems with Applications, 113, 186-199.

De Veaux, R. D., Agarwal, M., Averett, M., Baumer, B. S., Bray, A., Bressoud, T. C., ... & Ye, P. (2017). Curriculum guidelines for undergraduate programs in data science. Annual Review of Statistics and Its Application, 4, 15-30.

Parry, M. (2018). Data scientists in demand: new programs train students to make honest sense of numbers. The Chronicle of Higher Education.

Song, I. Y., & Zhu, Y. (2017). Big data and data science: Opportunities and challenges of iSchools. Journal of Data and Information Science, 2(3).

Tate, E. (2017). Data Analytics Programs Take Off. Inside Higher Ed.