Continuous Integration for Single Page Application

Samiksha Ghodke[1], Sharvari Joshi[2], Aashish Gore[3], Shreyas Karandikar[4], Prof.V.A.Kanade[5]

U.G Student, Department of Computer Engineering, PVG's College of Engineering and Technology, Pune, India[1,2,3,4]

Assistant Professor, Department of Computer Engineering, PVG's College of Engineering and Technology, Pune, India[5]

ABSTRACT: Failing of integration builds are show stoppers. Integration hell is a situation wherein code fails to compile or bugs arise which is extremely risky for a project and its success .These problems exceed the project schedule and exceed budget .The above situation occurs in software testing phase of software development life cycle which is the most difficult and expensive phase. Continuous integration plays a vital role against integration hell situations by frequently merging all developers working copies with a shared main branch.
With the advent of single page application which loads single HTML page and dynamically updates that page as the user interacts. This paper focuses on testing single page application by applying agile development methods such as continuous integration.

KEYWORDS:Integration, AngularJS, GitHub, Single Page Application,Jenkins

I. INTRODUCTION

Continuous Integration is a development practice requiring developers to integrate code into a shared repository several times a day .Each check-in is verified by an automated build, allows teams to detect problems early .Continuous Integration allows you to detect errors quickly and locate them easily. It focuses on reducing integration problems and allowing you to deliver software more rapidly.

In this paper,we are describing the continuous integration system which will be used for the deployment of single page application using Jenkins as the CI Server and GitHub as the source code repository .In the last decade, there is growth in the demand of web applications for business purposes. In order to ensure that web application conforms to quality attributes like security, reliability, scalability and accessibility,testing plays a vital role.

In traditional web applications,communication with the server is initiated by the client by requesting a page .server processes the request and sends the HTML page to the client .In subsequent interactions with the page,a new request is sent to the server and the flow starts again. But,in single page application entire page is loaded in browser after initial request and the subsequent interactions take place through AJAX requests This means browser will update only portion of page that has changed which avoids reloading of entire page which saves bandwidth as well as no loading of external files such as images or CSS files .This approach reduces the time taken by an application to respond to user actions, resulting in more fluid experience.[1]

We will build SPA with AngularJS to avoid issues faced by developers on using JavaScript,making the code complex. AngularJS divides front end into three parts, model, view and controller. This makes the code understandable to achieve high efficiency and good quality as well as leads to fluid user interface.

In this paper we explore, how the use of continuous integration system will bring about the deployment of Single Page Application (Financial analysis with graphical representation) using AngularJS which will help the user of the application to make a better and informed financial decision for investing in a particular MNC based on variety of criteria. This application will also have an added functionality for comparing the statistics of different companies which will help the user to make a wise decision.
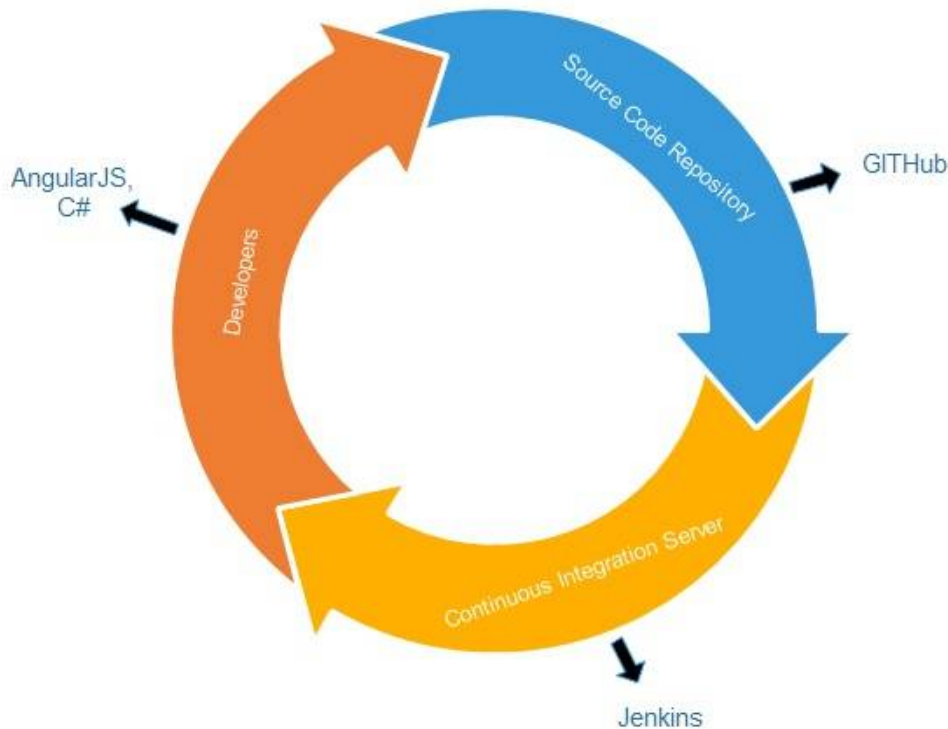
Fig.1.Continuous Integration Process for Single Page Application along with the tools used at different steps, AngularJS or C# at developing stage, GitHub as a source code repository and Jenkins as a continuous integration server.

II. RELATED WORK

1.Traditionally JavaScript was used as a frontend for providing static web pages to the client however AngularJS has taken over JavaScript as the maintenance and the development is less complex, as frontend of AngularJS is segmented into three components as follows-
Model- It is the actual data that is to be displayed.
View- In this the displaying of the data is done through templates.
Controller-It is the logic which glues the two components Model and View.
Moreover the controllers of Angular JS communicate with the server in the background and they are fully responsible for controlling the behaviour of Single Page Application (SPA). AngularJS also provides different testing modules as a result testing of the web application becomes easy.[1]

2.There are two well-known and commonly used agile practices Test Driven Development (TDD) and Continuous Integration (CI).In TDD the programs are developed incrementally where the tests are written first and then the code is developed to that extent where is can fulfill the tests required. Continuous integration in its simplest form, involves a tool which monitors the version control system for changes. As soon as the change is detected the tool automatically compiles and tests your application and all the developers engaged in the project are notified immediately[2]

3.Jenkins is a Java based Continuous Build System and is supported by over 400 plugins.Jenkins provides Source Code Management (SCM), Testing, Notification, Reporting, Artifact Saving, Triggers and External Integration.Jenkins provides various general options such as-Associating version control system.It helps in triggering the builds (Periodic, Polling)Sends notifications via EmailJavadocs and Junits test results are publishedShell scripts, bash scripts, Ant targets and Maven targets can be executed Jenkins executes the build in an executer where one executer per core is allocated on the build server, it also uses the concept of salve build servers where the load is distributed and is useful for building

on different architectures. As Jenkins supports external plugins and has a core feature set it can be an integral part of any Continuous Integration system.[3]

III.   PROPOSED SYSTEM

In the proposed system client-server architecture is maintained .On server side, continuous integration server, Jenkins will be notified about the changes when the user will push changes in the source code repository .It will either poll the repository every few minutes to see if any changes have been pushed or the repository will actively call the continuous integration server to let it know when the change is available. When continuous integration server sees the change it will clone the repository on the server or may instruct set of build servers to check out all the code and then build the project. If it fails it will send a notification to team.
On the client side user will check-in single page application code

Client Side:

AngularJS:Is an open source JavaScript framework maintained by Google and community which can help the developers to create single page applications.It helps in developing web applications with model-view-controller(MVC) capability in an effort to make development, maintaining and testing easier.SPA is getting popular nowadays and the technology like AngularJS aids to create such applications.

Server Side:

GitHub: Git is released under GPL's open source license. It is available freely over the internet.It is a distributed version control system. Clients of distributed version control system not only check out the latest snapshot of the directory but they also fully mirror the repository. Every checkout is a full backup of the repository. Git does not rely on the central server and that is why you can perform many operations when you are offline. You can commit changes, create branches, view logs, and perform other operations when you are offline. You require network connection only to publish your changes and take the latest changes.

Jenkins:Is a free, open source and most widely used tool for maintaining continuous integration cycle .It has got over 600 plugins to customize Jenkins as per you need.It mainly focuses on two jobs building or testing software projects continuously and monitoring executions of externally-run jobs.It catches problems fast and gives a rapid feedback.

IV.   EXPERIMENTAL RESULTS

Figures show the results of the single page application by using AngularJS. Figures 2,3 and 4 are given below.The single page application shows a list of companies viable for investment, based on filters selected by the user and shows dynamic changes in graphs, charts with respect to the filters.

Fig.2.List of companies and a map to select a country, this will allow user to select from list of companies and then dynamically give the detailed statistics in the form of graphs and charts.

Fig.3.Shows statistics of companies.

Fig.4.Financial analysis with graphical representation, the entire GUI will look like this on a web browser and by the magic of single page application there will be no page reloads. The will keep on updating dynamically as per user selections.

V.   CONCLUSION

In this paper, we have proposed a continuous integration system which will deal with many components including source code repository, an automated build system, tests and continuous integration server. The continuous integration server will automatically detect the changes in the application, run all the tests and build deployable version of the single page application with the goal of detecting problems early in the process. The next step will be to develop this single page application(Financial analysis with graphical representation) using AngularJS which will help the user of the application to make a better and informed financial decision for investing in a particular MNC based on variety of criteria. This application will also have an added functionality for comparing the statistics of different companies which will help the user to make a wise decision.

In a nutshell, our approach is to satisfy the customer with an error free application by using integration testing and to deliver the final product on time and help in keeping technical debt down. On successful deployment of the single page application reports, graphs and charts of the results will also be generated by the Continuous Integration system.

REFERENCES

[1] Madhuri A. Jadhav, Balkrishna R. Sawant and Anushree Deshmukh. Single Page Application using AngularJS. (IJCSIT) International Journal of Computer Science and Information Technologies, Volume 6 – Issue 3, 2015.
[2] Sheikh Fahad Ahmad, Mohd. Rizwan Beg andMohd. Haleem.Test Driven Development with Continuous Integration: A Literature Review.International Journal of Computer Applications Technology and Research Volume 2– Issue 3, 277 - 281, 2013.
[3] Jesse Bowes.Jenkins Continuous Build System. CSCI-5828.Spring 2012
[4]http://jenkins-ci.org/
[5]M. Fowler, "Continuous integration", http://martinfowler.com/articles/continuousIntegration.html, 2006, accessed June 2014.

[6] Bogdan Vasilescu, Stef van Schuylenburg, Jules Wulms, Alexander Serebrenik and Mark G. J. van den Brand. Continuous integration in a social-coding world: Empirical evidence from GITHUB.2014

[7] Sebastian Elbaum, Gregg Rothermel and John Penix. Techniques for Improving Regression Testing inContinuous Integration Development Environments. FSE'14, November 16–22, 2014, Hong Kong, China

[8] Tijs van der Storm. Backtracking Incremental Continuous Integration.

BIOGRAPHY

Mrs.V.A.Kanade is an Assistant Professor in the Computer Department, PVG's College of Engineering and Technology, Pune, India. She has completed her M.Tech. from Bharati Vidyapeeth Deemed University,Pune.

Samiksha Ghodkeis a U.G. student in the Computer Department, PVG's College of Engineering and Technology, Pune, India. Her research interests are Database management systems and Software Engineering.

Sharvari Joshi is a U.G. student in the Computer Department, PVG's College of Engineering and Technology, Pune, India. Her research interests are Database management systems and Data Structures.

Aashish Goreis a U.G. student in the Computer Department, PVG's College of Engineering and Technology, Pune, India. His research interests are Computer Graphics, Networking and Software Engineering.

Shreyas Karandikar is a U.G. student in the Computer Department, PVG's College of Engineering and Technology, Pune, India. His research interests are Data Structures, Networking and Software Engineering.