D3-Collab - Bridging the collaboration gap between designers and developers on the enterprise level

Ashish Durgude

UX Visual Designer, Red Hat, Bangalore, India

---***---

Abstract - *Collaboration between the designers and the developers is the most critical task in the attempt to roll-out a product in a given timeline. Despite of consistent efforts, the enterprises are still struggling to deal with this gap. Even after designers give a design handoff to developers, they need to rework on the UI coding rather than focusing on its backend functionalities and this process can be lengthy and iterative.*

This paper introduces D3-Collab, a comprehensive collaboration tool for designers and developers to collaborate effectively in the iterative process of the product development. This application will help designers to give a design handoff (to developers) as well as it will help developers to get the real-time design updates to replace only the specific part of the code where the changes are found during the analysis of the available versions of the design files, which are sent by the designer.

Key Words: Human-computer interaction (HCI), UI, UX, Visual design, Interaction design, Design & development, Design tool

1. INTRODUCTION

Creating visual designs and giving them to a developer to create a code can be very tedious while working with an iterative process. Nowadays, we are seeing so many applications like Zeplin[11], InVision inspect[10], Sketch plugins[12] which are helping developers to get the margins and some other CSS properties, but developers still need to do the front-end work. There can be 8-10 visual designs created for a specific product that developer needs to code, and at the enterprise level, this is not a one-time process, it's iterative and that depends on the feedback given at sprint levels as well as release levels.

In this paper, I am introducing an application concept that could potentially overcome all the problems listed above at a personal as well as enterprise level.

Human-computer interaction is all around us because of the expansion of technology in every domain. If you are a designer, then you have various applications like Adobe Photoshop[13], Sketch[12], Adobe Illustrator[13], Adobe XD[13], etc. at your disposal to expose your work to the world. If you are a developer, you have an array of programming languages, markup languages to learn and build your own applications. But it still doesn't cater to the listed conflicts that take place in case of excessive workload.

2. THE CURRENT METHODS OF COLLABORATION BETWEEN DESIGNERS AND DEVELOPERS

The research shows that currently, enterprises are using InVision[10], Zeplin[11] and sketch plugins[12] as main tools for giving design handoffs[1].

To get the design handoff from InVision, you need to upload the artboards of your PSD (Adobe Photoshop) or Sketch. Craft from InVision comes in handy while uploading the artboards directly from the software (ex. Adobe Photoshop and Sketch). After uploading the artboards, in the preview mode when a user clicks on the inspect menu, they can see the design handoff.

Sketch plugins make it possible to get design handoff directly from Sketch. Plugins like Sketch Measure[2] and Marketch[3] gives the exact same output as InVision Inspect.

The Sketch plugins and InVision Inspect give the CSS for particular elements. These design handoffs also provide padding, margins, fonts and color used just by clicking on the elements.

Zeplin[11] is different than these Sketch plugins just by one factor and that it facilitates collaboration. Unlike Sketch plugins, Zeplin[11] is a collaboration tool for both designers and developers. Once you log-in to Zeplin[11], whether you are a designer or developer you can add multiple members to the project. Unlike sketch plugin[12], you can comment directly on the screens.

These two features of Zeplin are also available for InVision [10] but not for InVision inspect. You need to add members to the complete InVision project to give them access to InVision inspect. There is a public share link service offered by InVision so that even if the end user is not a part of the InVision (not having InVision account), they can still see the screen and comments but the public link does not give access to InVision inspect. it is only restricted to comments.

Similar to Zeplin[11], a user needs an InVision account to access design handoffs.

Applications like RapidUI[8] and Supernova[9] doing an amazing work of creating code from PSD or Sketch file. RapidUI[8] generates code into HTML, CSS and JS and Supernova[9] generates code for mobile platforms like ios, Android and React Native.

3. HOW DOES THE CURRENT COLLABORATION METHODS MATTERS WHEN IT COMES TO ENTERPRISE LEVEL?

In HCI there are many lifecycle models including software engineering lifecycle models like:

- Star Life Cycle Model (HCI)
- Usability Engineering Model (HCI)
- Simple Interaction Design Model (HCI)
- Waterfall Life Cycle Model (Software Engineering)
- Spiral Model (Software Engineering)
- RAD Life Cycle Model (Software Engineering)
- DSDM Life Cycle Model (Software Engineering)

Enterprises follow the lifecycle models which suits best their work environment and mostly follow the application lifecycle management (ALM) process. Products like Openshift.io, Visual Studio Online follow agile methodologies that maintain and track workflow.

When it comes to design & develop stage, eventually it becomes an iterative process. Whenever designers submit their UI design work to the enterprise - the internal feedback, user feedback, UI changes, UI expectations, sprint end, release end, all of these factors related to the process come into the picture.

Enterprises usually have more than 10 to 12 screens at a time for a project for the designers and developers to work on. One single developer or designer does not work on the whole project; there are teams which divide the workload among themselves. In such a situation, team communication, work management and collaboration are critical to remain on the same page.

4. WHY ARE CURRENT COLLABORATION METHODS UNABLE TO PROVIDE BETTER COLLABORATION WHEN IT COMES TO ENTERPRISE LEVEL?

If you follow any of the HCI life cycle models mentioned above, it's a proven fact that when design and development follow the iterative process, they end up giving better results than with every iteration.

Let's take an example of Star Life Cycle Model,

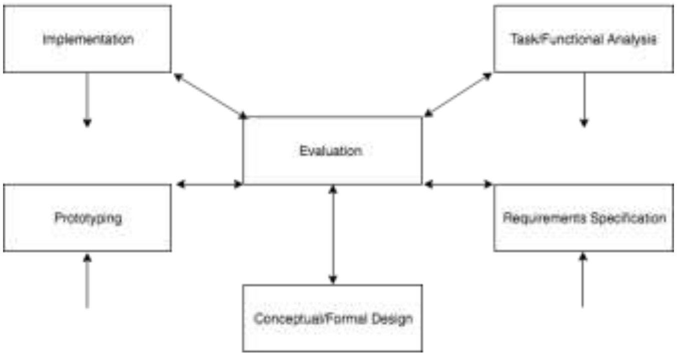


Fig -1: Star Life Cycle Model

Star life cycle does not specify any activity order except that you need to go through the evaluation first. Evaluation is central to Star life cycle model. It does not matter which activity you are performing, whenever that specific activity is completed, you need to do the evaluation of the result of that activity.

4.1 Evaluation Types

Evaluation can be performed in 3 ways,

- Qualitative Evaluation
- Empirical Evaluation
- Predictive Evaluation

Qualitative evaluation emphasizes the totality of a phenomenon.

Empirical evaluation is based on the numeric summaries or observations of the phenomenon.

Predictive evaluation is based on the systematic application of pre-established principles and heuristics.

4.2 Competitive analysis based on enterprise aspects

At the enterprise level, submitting a design and performing its evaluation is not a one-time process. The multiple iterations happen before the final design goes into development. This can be followed by another complete cycle of design and development as well.

Now let's come back to the applications mentioned in the section of "the current and upcoming state of collaboration between designer and developer".

InVision[10], Zeplin[11] and Sketch plugins[12] will give design handoffs in the format like margins, paddings, and some basic CSS code. These applications help developers to a certain level. The developer still needs to do the coding in HTML, AngularJS or whichever language they are using to develop.

RapidUI can only export in HTML, CSS and JS format. It does not have support for other programming languages like AngularJS or any other available JavaScript framework. It only supports basic CSS animations and styling. RapidUI[8] is very useful for personal use but not for enterprise. If there is a company that only needs some basic HTML, CSS based UI for company's portfolio, or some other work, then RapidUI comes in handy, but for a company that wants more than simple UI and has enough number of developers to work on it, then RapidUI[8] will not work. Developers will invest their time in coding rather than going to a different application just to get CSS and JS.

Supernova[9] is an application which exports the code only for mobile platforms like ios, Android and React Native. This makes it a good application for mobile development but not for web UI and front-end frameworks.

The common problem with these applications is that they retain the data, which is not preferred for enterprises. Another big concern is the iterative design process inside the enterprise. If you build an application using one of these applications and export the code to your local repository, and if some UI changes come after that, then you again need to create the application from the start which is very frustrating.

Pix2code[4,5] is the existing research project. The problem with Pix2code is that it accepts only one image at a time to produce the code. Uploading single screen every single time and exporting code can be a very lengthy process and too much rework.

5. FINDINGS OF THE KEY PROBLEM DURING RESEARCH

After doing research on current applications in terms of designer and developer communication and collaboration, six key problems have been identified:

1. Limitations on the export of design handoffs and code
2. Real-time customization is not possible
3. Failure in keeping up with an iterative process
4. No direct communication between design and development
5. Semi-free applications (except pix2code)
6. No real-time updates for changes

6. INTRODUCING D3-COLLAB

D3-Collab is a research project which will help designers and developers to collaborate effectively. D3-Collab stands for "Designer-Developer-Direct Collaboration". It is made keeping in consideration the enterprise aspects. To dig deeper, let us look at D3-Collab in reference to visual design.

6.1. The main purpose of D3-Collab

In the competitive analysis, it was observed that all the application are using some techniques to either convert the visual design to code or directly export the code based on user's action in website builder applications. D3-Collab uses machine learning to convert the visual design to code supporting multiple programming languages, but that is not the main purpose of the D3-Collab.

Its main purpose is to build a better collaboration between designer and developers during a complex iterative process in enterprises.

As we know, in the enterprise, the project is going to have many iterations and front-end developer needs to work on code and change code multiple time based on UX input. D3-Collab solves this problem. If some changes come by UX in visual design (UI), then D3-Collab will only change the specific part of the code where change is needed so that developer doesn't have to worry about iterations in visual designs.

6.2. How D3-Collab works?

In D3-Collab, it is important that designer uploads their designs by specifying the version of the followed iteration. The version is identified by the application to convert a specific part of the visual design into code. A detailed workflow of D3-Collab for designers and developers is explained in the upcoming sections.

Creating the versions of the designs is the key factor in D3-Collab. Designers need to upload the designs in the versions. These versions are nothing but the stack which includes a set of values assigned to the screens uploaded in the one version. Each value will have an ID associated with it. When a developer uses this application, he will request the designer for an integration link and based on that he will automatically get the designs and then he can convert it to code. Let us assume a designer uploads a design in "Version 1". Now suppose that the new design changes are made to it. Those changes need to be uploaded in D3-Collab as "Version 2". Then the designer will send integration link to the developer. When the developer clicks on the version 2, the two available versions will be compared. It will check for differences between the set of values associated in the stack 1 (Version 1) and stack 2 (Version 2). After that D3-Collab will check for the code written for Version 1 and Version 2. It will analyze the changes using mathematical models used in machine learning. After analyzing, it will give an option to change specific parts of the code which will not affect any other part. This code is generated by D3-Collab. If the developer agrees to apply changes to specific part of a code, then it will replace that code automatically so that developer doesn't need to do it manually.

Below are the images to show the version concept in a visual way:

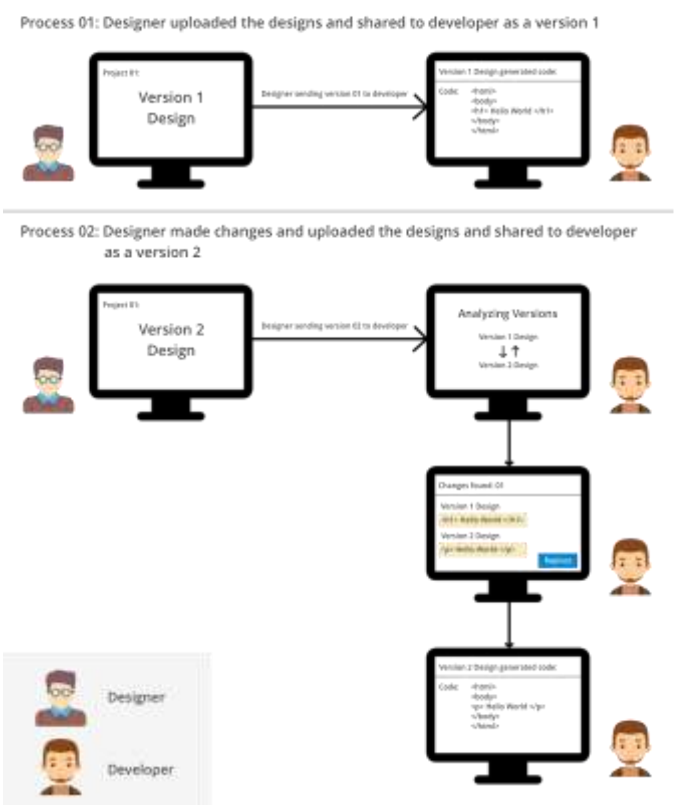

Process 01: Designer uploaded the designs and shared to developer as a version 1
Project #1
Version 1 Design
Designer sending version 01 to developer
Version 1 Design generated code
Code: <html> <body> <h1> Hello World </h1> </body> </html>
Process 02: Designer made changes and uploaded the designs and shared to developer as a version 2
Project #1
Version 2 Design
Designer sending version 02 to developer
Analyzing Versions
Version 1 Design
Version 2 Design
Changes found: 01
Version 1 Design
<h1> Hello World </h1>
Version 2 Design
<p> Hello World </p>
Approve
Version 2 Design generated code
Code: <html> <body> <p> Hello World </p> </body> </html>
Designer
Developer

Fig -2: A visual flow of D3-Collab for designer and developers

7. USER SEGMENTATION

So far we have only looked at designers and developers as stakeholders, but the user base for D3-Collab expands beyond that. Managers and reviewers also come into the picture. To categorize in terms of roles and operations, there are two kinds of users for D3-Collab:

7.1 Primary User(s)

The primary users are developers and designers.

7.2 Secondary User(s)

The secondary users are project managers and reviewers.

7.3 How the user segmentation for D3-Collab is done?

This project is mostly focused for a developer needs to create better products in less time, although, without the designer's consideration and inputs, this project will not work. This project will help developer to focus on building application functionality rather than worrying about the UI integration. It will not replace designer or developer in any way. It will be a huge help for a developer for delivering the code within the timeline set by the enterprise in the form of sprint end, release end, etc..

In enterprises, managers and reviewers help during the process of a project. They check and review the designs that designer has created. These two users will be secondary users because they will not be interacting much on this platform.

8. USER FLOWS FOR D3-COLLAB

As mentioned in section 7.1, the primary users are designers and developers. Each user is related differently to D3-Collab. Designer will upload the designs and share it with developers and developers will take the design and convert the code using D3-Collab. The process for designers and developers will be explained in the upcoming sections. In below images, I am showing user flows for both designer and developer to show how they are connected with each other in D3-Collab.

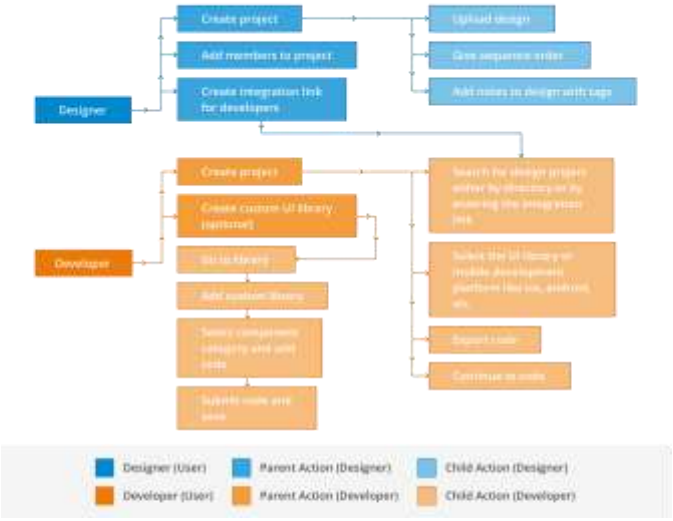


Fig -3: User Flow 01 of D3-Collab for designer and developers

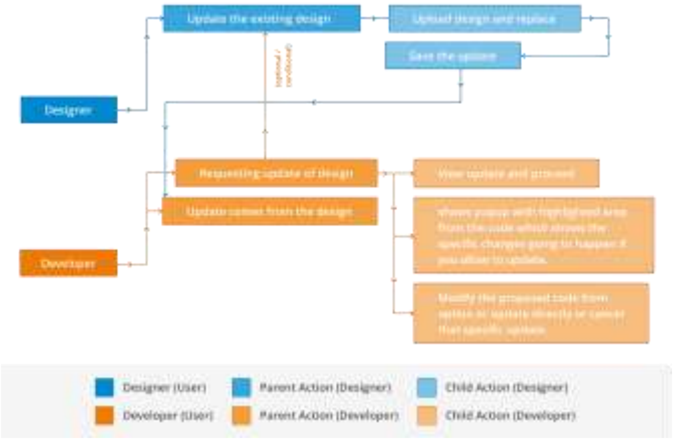


Fig -4: User Flow 02 of D3-Collab for designer and developers

As mentioned in section 7.2, there is a secondary type of users: Managers and Reviewers. They are not going to look for code conversion or how many versions have been created. They could leverage this application for reviewing designs and putting their comments/demands for changes on them. They have nothing to do with the developer view of an application.

Below I have attached one user flow for the secondary users

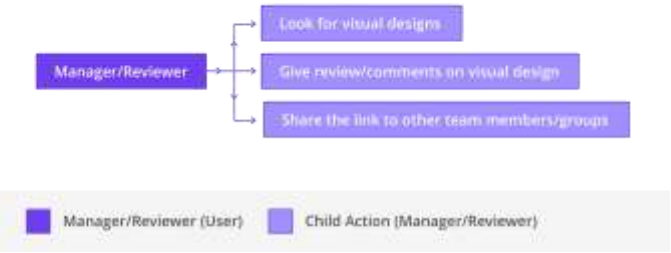

Manager/Reviewer
Look for visual designs
Give review/comments on visual design
Share the link to other team members/groups
Manager/Reviewer (User)
Child Action (Manager/Reviewer)

Fig -5: User Flow 02 of D3-Collab for managers and reviewers

To show how the D3-Collab look like and how the designers and developers will use this application, I am giving step by step details using visual designs that I have created for D3-Collab.

9. D3-COLLAB APPLICATION UI

There are two different types of UI for designers and developers. Both users will use D3-Collab differently. The first UI is a web application which is made for designers, managers and reviewers. The second UI is an extension integrated with code editor which will be used by developers. The visual design presented in this paper is for reference purpose only.

Let us start with designers UI view.

10. D3-COLLAB FOR DESIGNERS

D3-Collab is a web application in which designers can upload the PSD or Sketch file for visual designs in formats like png, jpeg, etc.. Designer can create versions on those designs and share it with developers. They can add notes for the developers on the screens. After/before attaching the files, designers can add the members. Those members can be a group of designers, developers, managers and reviewers.

10.1 Dashboard of D3-Collab

The dashboard is the first view where a designer will be able to see their recent project, the members associated for that project, any new comment on a project, etc. there is a button named as “create new project” by which they can create a new project.

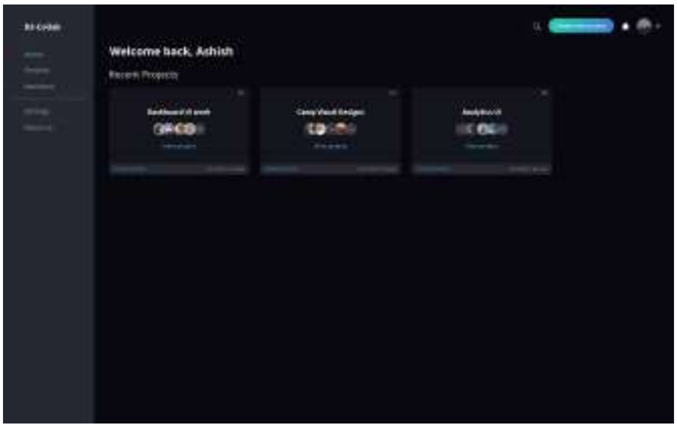


Fig -6: Dashboard (designers view)

10.2 Create a new project and adding members in D3-Collab

Once a designer clicks on “Create new project”, a modal will appear. In this modal, a form is presented which includes a name of the project, option for adding members, etc.. After a user fills the form, they can confirm their action for creating a project. Then a blank project will be created where they can upload their designs.

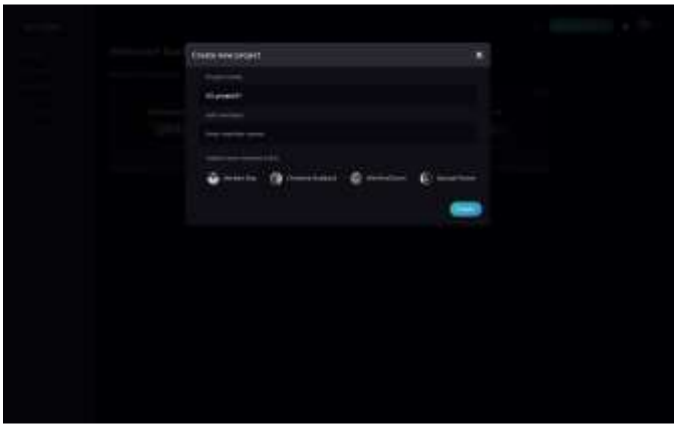


Fig -7: Create new project modal

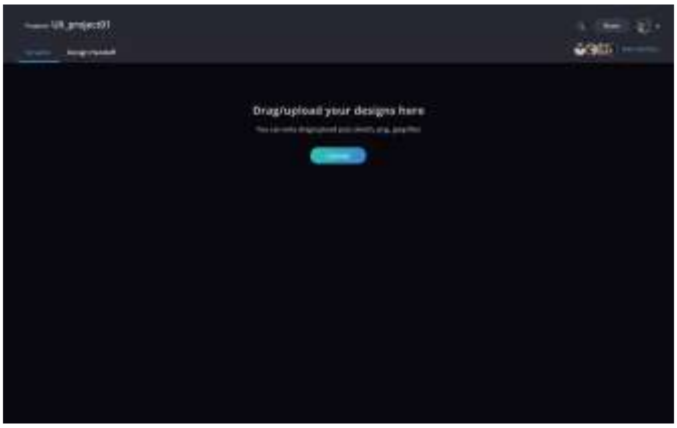


Fig -8: New project empty view

10.3 Uploading new screens and sharing the design project with developers in D3-Collab

After uploading the screens, a designer can share the project by clicking on "Share" button. After clicking on the share button, a modal will appear where they will have two options to share. They can either create a public share link which could be shared with other team members, managers, reviewers, etc., or they can also create a link called as "Integration link" which will be used for sharing with developers so that they can work on the code.

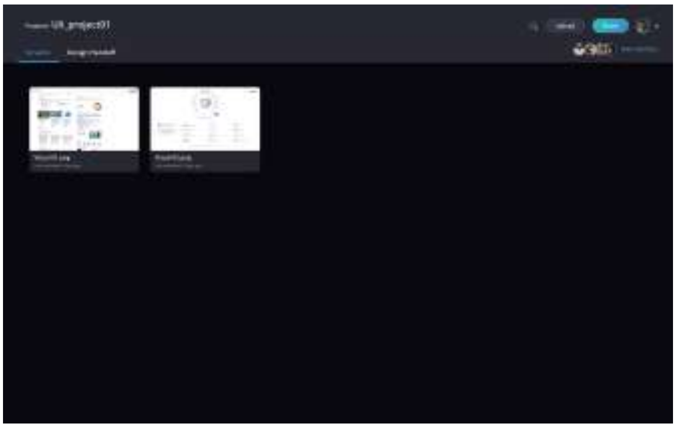


Fig -9: project view with images uploaded in it

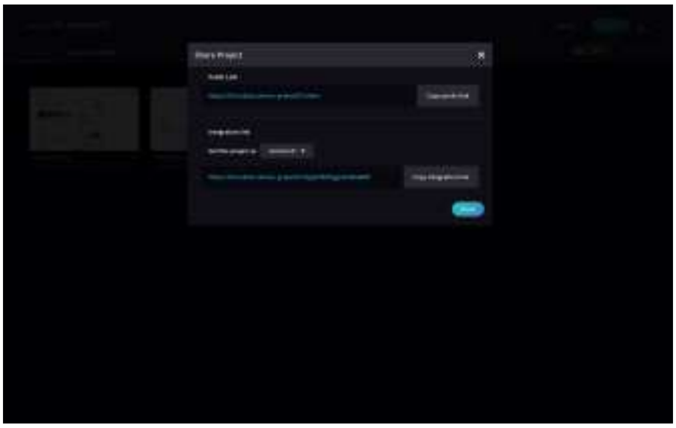


Fig -10: Share project using the public link and integration link

10.4 What is integration link?

Integration link is the connection between the designer's input and developer's dashboard view. A developer will get the access to the file from the designer by entering the integration link. Under normal circumstances, when a developer opens the integration link in the browser, it will give the overall view of the files that are presented in that specific project. The developer can also see design handoffs for screens. They will be able to see different versions for specific project uploaded by designers. A developer can convert a visual design into code after getting access to integration link.

11. D3-COLLAB FOR DEVELOPERS

Most of the developers do not like to use multiple applications at the time of coding. They usually use one specific code editor to code. Developers at Red Hat prefer to use VScode[14] for coding, which is going to be used for explaining certain cases in the forthcoming sections.

The reason behind choosing VScode[14] is that it supports extensions, hence it would be a wise idea to create a D3-Collab extension for VScode[14].

There is one separate section in VScode[14] for D3-Collab where the developer will get the notification when designer updates some design in D3-Collab web application. Also, a developer can see the visual designs in VScode[14]. In this view, the developer can decide whether they want to convert the visual design to code and make changes in code based on the different version of design or not.

To get more understanding of developer view of D3-Collab let's go through the visual design flow.

11.1 Dashboard of D3-Collab inside VScode

Below screen is the dashboard view of a D3-Collab extension for VScode[14] in which developers can see all the projects and versions included in it. This UI is divided into 3 parts. The first part is on the left side where a developer can see project names, version names, screens count for a specific version. The second part is on the middle where a developer can see the actual visual designs presented in a specific project. The third part is on the right side where developer can work on code. Also in this third part, the code will get generated after converting visual designs.

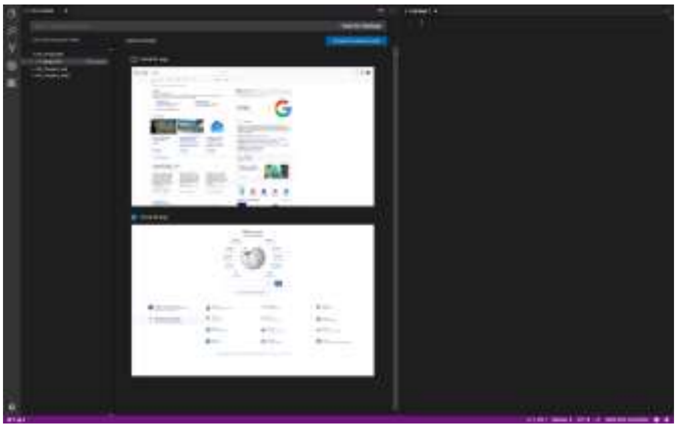


Fig 11: VScode with Active D3-Collab extension (developers view)

11.2 Converting design to code inside VScode

Developers can select the images they want to convert into code. Once developer clicks on "Convert (1) screen to code" button, a modal will appear where they could view different platforms to convert visual designs. These platforms are divided into two sections. The first section is "Web" where developer can select options like HTML/CSS, Bootstrap framework, Angular, React, etc. for converted code. The second section is "Mobile" where developer can select options like Android, ios, React native, etc. for converted code.

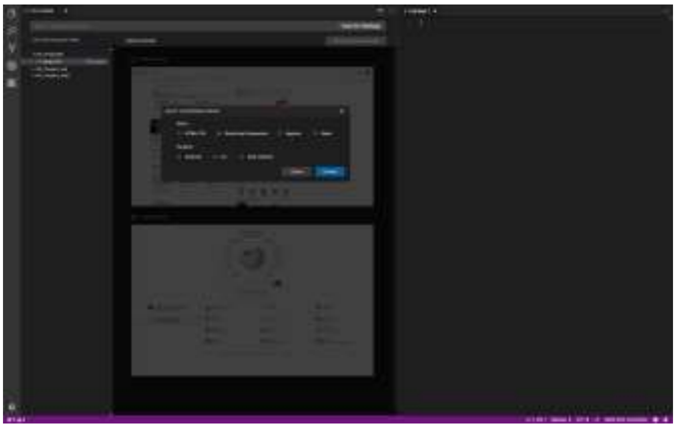


Fig 12: Select output screen

After selecting the output format for code, it will start processing and the code for that screen will be shown on the right side.

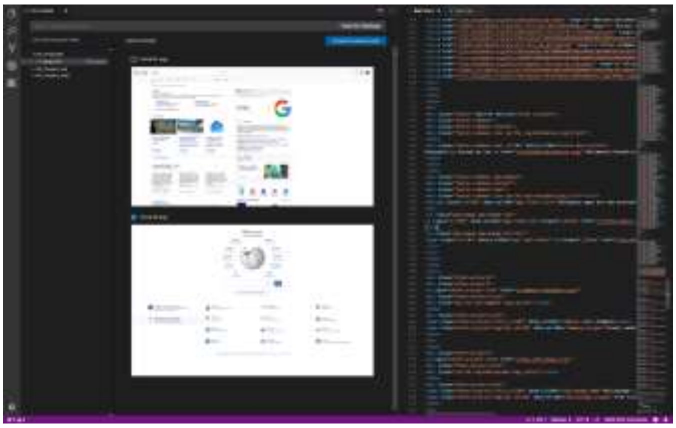


Fig 13: Converted into a code view

11.3 Receiving a new version of design inside VScode

If a designer made some changes in design after developer worked on the code, then the designer will upload screen in D3-Collab web application by setting the next version and then they will update the integration link. This update will be received as a notification inside the D3-Collab extension of VScode[14] which will show that there is next version available for a particular design project. It will also show how many screens are new, how many screens are changed/updated, etc..

If some changes found in particular screen, then that screen will be highlighted by yellow border and developer will be able to see the changes of a screen from version 1 to version 2 by clicking on "View change(s)".

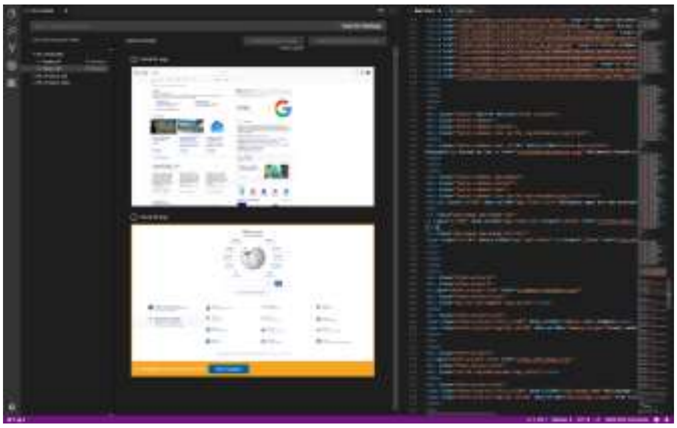


Fig 14: Received new version view

11.4 Checking the changes found in next version of the design

If there are any new changes found by D3-Collab for any specific screens, and a developer wants to see those changes then they can simply click on "View change(s)" button, after which a modal will appear where they can see a previous screen and changes made in that screen.

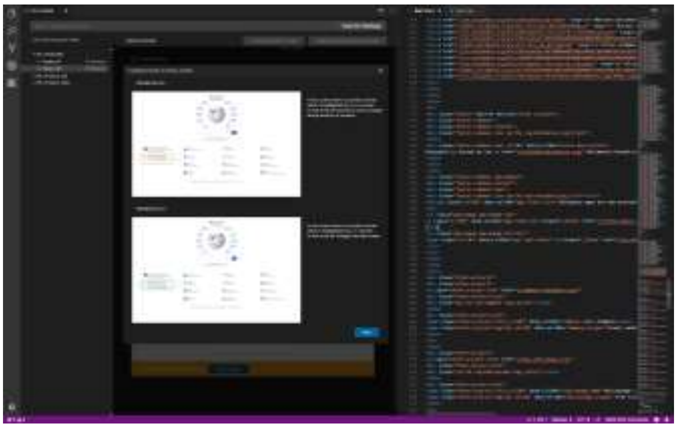


Fig 15: Changes found in different versions view

11.5 Converting to code by considering changes/updates in screens

When the developer selects a screen for conversion, they will get 2 options:

1. Convert to code with updates

2. Convert to code without updates.

If a developer decides to convert code by considering an update, then D3-Collab will replace code and that code will not affect the existing code.

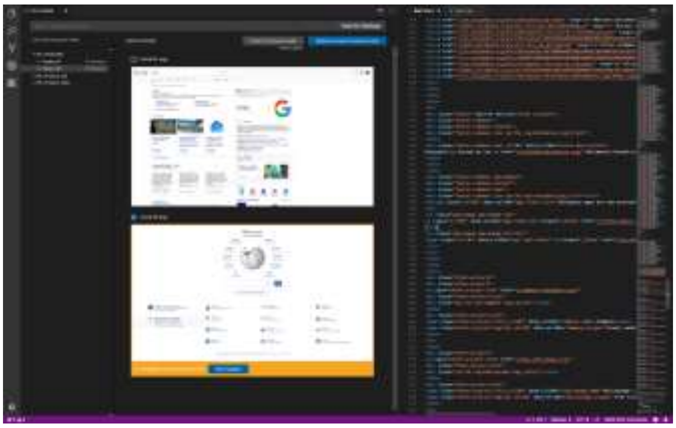


Fig 16: Selecting screen to convert it into code including updates

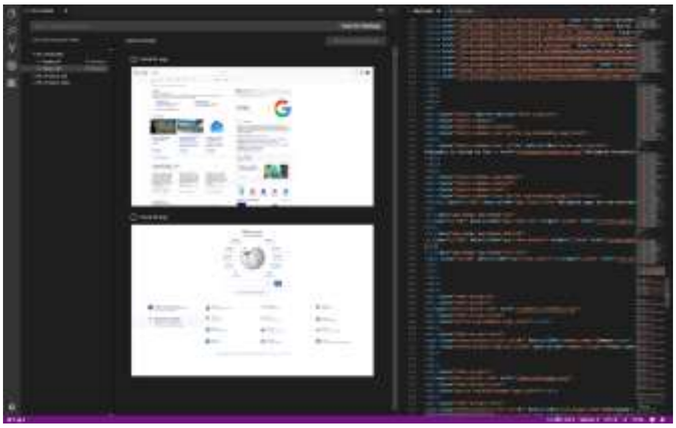
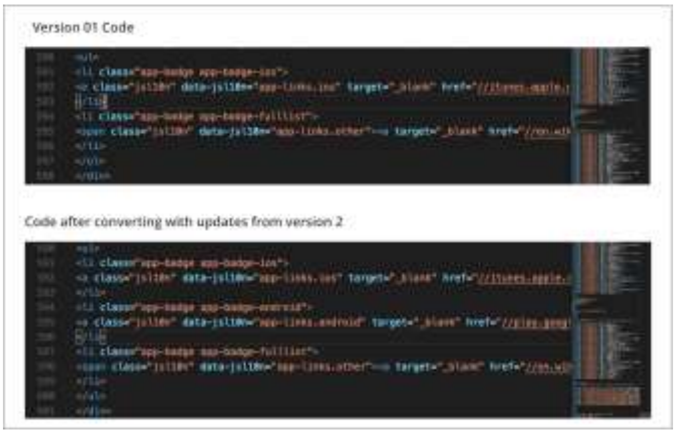


Fig 17: View after code conversion.



Version 01 Code

```
<ul>
<li class="app-badge app-badge-ios">
<a class="jsl10n" data-jsl10n="app-links.ios" target="_blank" href="//itunes.apple.
</li>
<li class="app-badge app-badge-fulllist">
<span class="jsl10n" data-jsl10n="app-links.other">→ target="_blank" href="//en.wi
</li>
</ul>
</div>
```

Code after converting with updates from version 2

```
<ul>
<li class="app-badge app-badge-ios">
<a class="jsl10n" data-jsl10n="app-links.ios" target="_blank" href="//itunes.apple.
</li>
<li class="app-badge app-badge-android">
<a class="jsl10n" data-jsl10n="app-links.android" target="_blank" href="//play.goog
</li>
<li class="app-badge app-badge-fulllist">
<span class="jsl10n" data-jsl10n="app-links.other">→ target="_blank" href="//en.wi
</li>
</ul>
</div>
```

Fig 18: Changes in Code

11.6 Viewing design handoffs using D3-Collab web application

If a developer wants to see the design handoff, then they can open the web application and under the "Design Handoff" tab, they can view all the specs by clicking on elements like a card, button, text, etc.. Here developers can also copy basic CSS code.

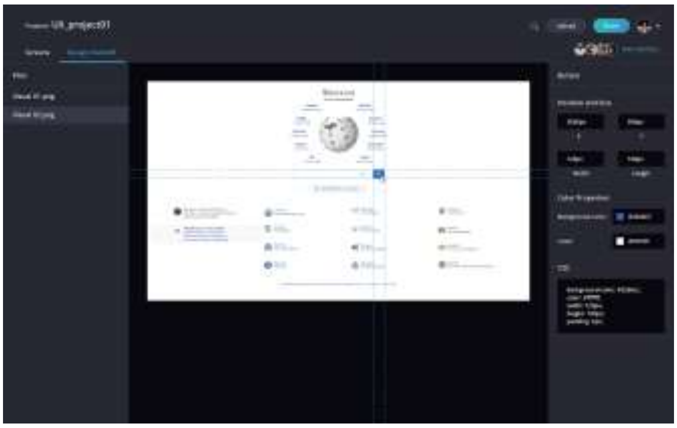


Fig 19: Design handoff view of D3-Collab web application

12. D3-COLLAB FOR MANAGERS AND REVIEWERS

The managers and reviewers will use the same D3-Collab web application which is created for designers for reviewing and commenting on the screens. They can add members to project and also share the public link with other team members.

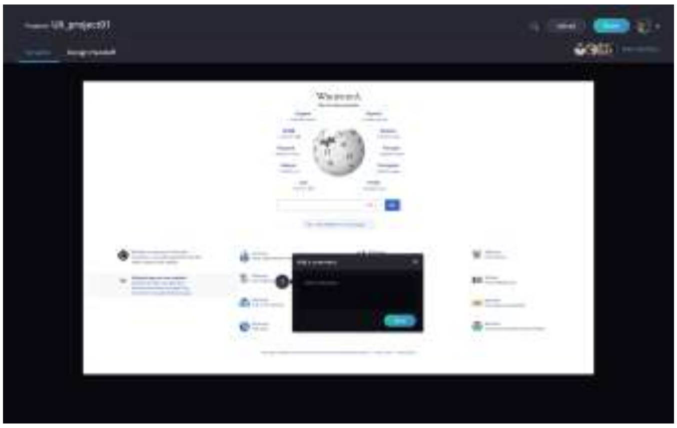


Fig 20: Add comment view of D3-Collab web application

13. CONCLUSIONS

The paper describes how D3-Collab can help designers and developers for building effective collaboration in iterative design process situation. It explains the importance of evaluation through Star Life Cycle Model.

A competitive analysis has been made for existing applications for understanding a current state collaboration between designers and developers.

In this paper, there are three user flow diagrams for the D3-Collab which explains the different scenarios of D3-Collab for designer, developer, manager and reviewers.

A brief description of the designer's and developer's role for D3-collab has been theoretically explained supported by visual designs.

By using this application, developers can easily convert visual design to code or update the existing code by changing only a specific part of the code which will not affect the existing code. The changes in version 1 and version two are found after analyzing the design files associated with it.

14. LINK TO HIGH-RESOLUTION IMAGES USED IN THIS PAPER

Follow the given link below to see images in a larger scale:

https://drive.google.com/drive/folders/1vpRfoJ4R_j8-WxM5f6TOm6HcmTMFGv8O?usp=sharing

REFERENCES

[1] 2017 annual design tool survey https://uxtools.co/survey-2017

[2] Sketch measure (sketch plugin for design handoff) https://github.com/utom/sketch-measure

[3] Marketch (sketch plugin for design handoff): https://github.com/tudou527/marketch

[4] Pix2code https://github.com/tonybeltramelli/pix2code

[5] Tony Beltramelli. September 19, 2017 pix2code: Generating Code from a Graphical User Interface Screenshots https://arxiv.org/pdf/1705.07962.pdf

[6] Pix2code Execution 01 https://www.youtube.com/watch?v=pqKeXkhFA3I

[7] Pix2code Execution 02 https://www.youtube.com/watch?v=DapKYJ6niCM

[8] RapidUI https://rapidui.io/

[9] Supernova https://supernova.studio/

[10] InVision https://www.invisionapp.com/

[11] Zeplin https://zeplin.io/

[12] Sketch https://www.sketchapp.com/

[13] Adobe Creative Cloud https://www.adobe.com/in/creativecloud.html

[14] Visual Studio Code https://code.visualstudio.com/