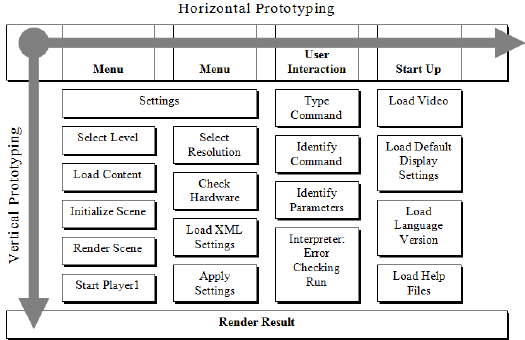
*Forms of Prototyping*

Prototyping is the initial stage of software release to either testing or just your design. It is the development evolution and product fixes that occur before the final release of the product. Most of the time it is called the beta phase or beta testing. Where the project will get evaluated in each stage and improved according to the client’s needs if need be. Another principle in prototyping is that it can be used to show others the current stage of the project to the client or stakeholders and to make sure you are still on track to complete the project. Some examples of prototyping are:

*Horizontal Prototype*

A Horizontal prototype is an outer shell of the entire system for example. Menus, Screens, Reports, Windows these all have little to no processing behind them. They typically contain all of the systems functions on the menus, it includes dummy screens, reports and databases for core functions. The Horizontal prototype contains no processing logic behind the external features. Later on the Horizontal prototype can be used to mould the start of the final system and can be developed into the final system

*Vertical Prototype*

Vertical prototyping contains a stripped down version of core functions within the system. These functions provide the ability to enter data and store it in to a database. It is also having the ability to display data on the screen of the user as query’s and reports. Vertical prototyping is used was a certain feature is complex within the system and it is poorly understood and to be explored and broken down

*Feasibility Prototyping*

The feasibility prototype proves some technical assertion that is key to the feasibility of the alternative. It verifies that the critical components of the technical architecture integrate properly and are capable of meeting the business needs. The information gained from feasibility prototyping will help to determine whether features are carried forward, discarded or if there is a need to revise/rework them.

*Concept Prototype*

A concept prototype is a high level application that give the views and users a vision of the overall project with functionality, design, structure, and operational characteristics of a system or project. The concept describes the required look and interaction of the interface, scope and topology. This prototype may be completed on either paper or online or even a combination of both some things are better shown or described on paper with a quick draft and they can do a final design online. The concept prototype is developed in the early stages of the system defining the architecture of the system

*Throwaway prototyping*

Throwaway prototyping is a part of the system which is developed and then given to the end user to test and evaluate. The user will provide them with feedback of the system and this can be quickly incorporated into the development of the main system. The prototype that was once used will be discarded or thrown away and a new one will be developed off of the new feedback from the end user. The main objective for the throwaway prototype is to make sure that the system being built fits all of the requirements that the user wants.

*Evolutionary Prototyping*

Evolutionary prototyping is in simple terms building on an earlier prototype that you may have created earlier. The prototype is built on by using the feedback from the user. So the developer will use this information to build on his current prototype hence the term Evolutionary.

*End User Testing*

End user testing is when the developer has his final project and it is all completed and now has to test what he has created to the users asking requirements to make sure that they are giving the user what they have asked for. Also they will test the program to make sure there is no errors and bugs they will do this via out putting betas for people to test and all of this will be logged with a Test Plan. Each test would be numbered and have a description of what is being tested. Then the expected outcome of what is meant to happen. Then the actual output what actually happened did it do the correct output.

Alphas – Alpha testing is a type of testing performed to identify all possible issues/bugs before releasing the product to everyday users or public with the further testing. The point of the alpha testing to keep the testing in house with the company that is rolling out the product. They are searching for any type of breaking within the system that can be taken advantage of by the public if got into the wrong hands. So, to keep it safe they normally only distribute alpha products to invite only or in house

Betas – Beta Testing of a product is performed by users of which are normally large forms with in the public rolled out in the waves of the software application in a "real environment" and can be considered as a form of external User Acceptance Testing. This will be the last form of testing before the product or system is rolled out as a finished project, so it is important that they get all of the necessary work done with the feedback from the beta and other testing

*Advantages and Disadvantages of End User Testing*

Firstly, the advantages of user testing are the fact that you can see where your project is currently at by listing the different needs that the user wants and see if you meet every one. If not, then you know that your project is not ready to be shipped to the customer. If you do this testing early enough you will leave your self-enough time to fix all of these changes if need be. Another benefit of user testing is you may find a better way or more efficient way to do something which can save space and do something better than what you had. Thirdly it gives you a chance to double check your work for any spelling errors or miss placed objects that can cause problems to the users. Finally, user testing can also help provide evidence for work that you did to be able to show the customer to show your worthy of the pay that you ask for.

A disadvantage of end user testing is the time that it takes to complete the feedback process. Most times you had it out to more than one person as well so if one person does not give you feedback who you were depending on then you are missing some vital feedback. Also it can cost money to roll out testing like alphas and betas which some companies don’t have the funds to do causing their games to not be tested correctly so when they come out they don’t do as well.

*Tools in Prototyping*

*Low and high fidelity*

Low-fidelity prototyping is a simple and easy translation of the product and design concepts. It’s used to turn the design ideas into testable and tangible artefacts, collecting and analysing the user demands at the early stage to later put into end user testing to create improvements.

High-fidelity prototyping is an interactive prototyping which is quite close to the final product, with lots of functionality and details included. High-fidelity is often used in the later usability evaluation to discover the potential issues that may exist in the workflow, interactivity and so on.

*Balsamic*

Balsamic is a user interface mock-up and website wireframe builder application web-based mock up. Its job is to allow designers to arrange pre-built widgets and tools to complete what is needed by the user to create what they need in the prototyping stage.

*Wire frame*

Wireframe is a low fidelity representation of a design. It shows the main groups of the content that will be within the project and where all of the information will be on the screen / document basically a structure of the information and how it will be displayed to the user. Wireframes should be created quickly and almost the whole time should be spent communicating with team members. The mere activity of wireframe-creation should be really quick. Wireframes are typically used as the documentation of the project. Since they are static and fix interaction with an interface at a specific point in time

*Mock ups*

Mock up is the visual script used to present the overall visual design of the product. It has a richer visual element than wireframe, including graphics, layout, colour, and other more detailed visual presentation. To some degree, it's the final design of the product. Mock up is static and inoperable. It focuses on the appearance of the product through adding a wealth of visual elements to reach in high fidelity.

*Paper based*

Paper based is all about a pen and paper that the user will use that allows the user to quickly write or sketch ideas, facilitating the internal discussion between group members.

*End User Feedback and Prototypes*

With the three different prototypes that was made though out my creation of my web-based game. I had a lot of feedback from peers within my class. Therefore, I had so many prototypes but was able to compress them down into three with the final one being finished. With the user feedback I took it in and saw what I could improve to make it fit the problems that was there and get rid of them. To Improve the game when final release comes.

*Prototype #1*

The first prototype was just to get the canvas on the screen to see the amount of room we were working with and to see the coordinates to implement the player and NPC.

One piece of feedback from the first prototype was to get the objects on the screen and getting them to work. By chasing each other and to get the collision down.

If you then proceed to look at the second prototype you can see that these things were implemented and added to the game.

*Prototype #2*

Looking at the second prototype you can see that the NPC square now chases the users square which the user controls by moving the mouse around in the canvas, but the squares are restricted to the canvas size to stop them running off the end of the screen.

Another piece of feedback was to see how many lives you have left once you collide with the NPC and to display them so that the person playing the game can see how many they have left.

Also, to display the current location of the mouse in coordinates on the top of the screen so that the user can keep track of their current position

*Final Prototype*

With the final prototype which is the finished game I took all the feedback into consideration and added in the lives at the top of the screen with a decent sized font so that the user can read it. Also, I added in the mouse coordinates at the top of the screen also so that they user can see the current position of their mouse.

*Feedback*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name of User | Test Description | Expected Outcome | Actual outcome | Feedback |
| Sam Pearce | Can the squares leave the canvas | For the squares to get locked within the canvas till the mouse returns | The Squares did get stuck inside the squares till the mouse returns | N/A |
| Luke Shead | Does the game end after you lose 3 lives | For the game to end once lives hit 0 | The lives are not tracked so the game does not end | This will be fixed later down the road as another prototype |
| Sam Dearing | Does the user know when he collides with the NPC | For the colour of the players square to change colour | The square does in fact change colour from green to red when it collides with the NPC | N/A |