

PROJECT PLAN REPORT

TEAM 3

BATCHANGARI ROHITH CHANDRA GOUD

BHARGAV VECHA

VINAY KUMAR REDDY KANDULA

PRODUCT NAME: PRIVATE CHAT BOX

PRODUCT DESCRIPTION

Development of a mode of communication between users, by developing a chat box in order to provide privacy. The end user will have a user interface using which they can respond and create messages to be shared between each other. Technical capabilities required to perform this task are, having knowledge of Java Script, CSS (Cascading Style Sheets) and using Microsoft Visual Studio.

The potential audience of this product can be, students of a class, a group of friends, strangers who wants to share their insights on a topic and for users wanting to share messages or communicate. This software will provide users with a chat room where about 25 people in the same time can communicate with each other using text and emojis.

The most important feature of this product is that, it can be used without providing a mobile number, E-mail address or any social media account information unlike most of the application that are found on web. No personal information is shared on the internet and is known by the peer user as privacy is one of the most concerning issue in today's world. The users can join a personal chat room by entering a specific key shared among them, to communicate and the whole chat will be deleted once they leave the chat session.

TEAM DESCRIPTION

This product is developed using Java Script, where all the backend process is developed using NodeJS and frontend will be developed using ReactJS along with CSS, therefore the team members should exhibit knowledge in coding JS and also using Microsoft Visual Studio which will be used to write the code. The team members should also have knowledge in using Modified Waterfall Software Process Model, testing and debugging the software. Strong motivation, teamwork and communication between team members will ensure successful development of the software.

SOFTWARE PROCESS MODEL

**Software Process Model:**

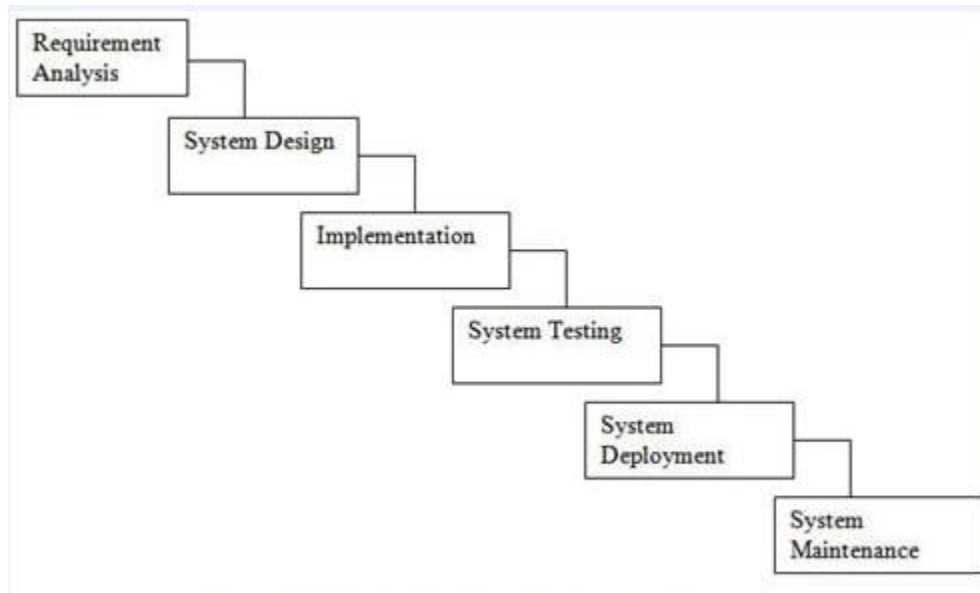
A software process model helps us follow a systematic procedure in developing software. The general process steps are:

- Requirements
- User Interface Design
- System Design
- Program Development (design and coding)
- Acceptance and release.

We will be using the Modified Waterfall Model.

### **Waterfall Model:**

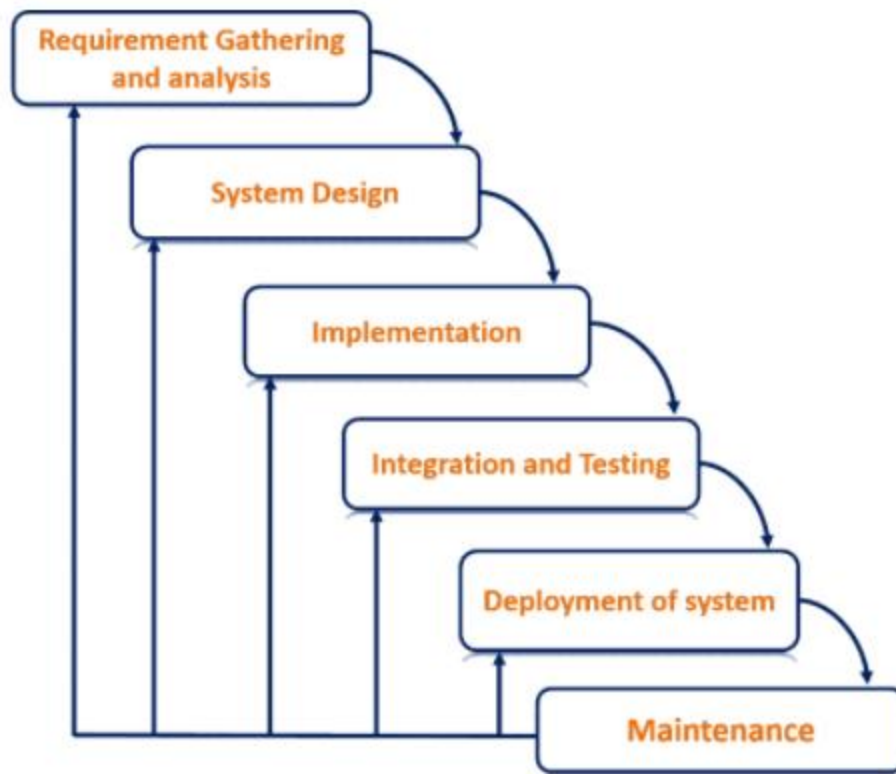
The waterfall model is a heavyweight process with full documentation of each process step. It is easy to understand and use. In waterfall model, each phase must be completed before starting the new phase. This model illustrates the software development process in a linear sequential flow. In the waterfall approach, the whole process of software development is divided into separate phases. In this Waterfall model, typically, the outcome of one phase acts as the input for the next phase sequentially. The following are different phases implemented in the waterfall model.



The disadvantage of waterfall model is that it does not allow much revision. Once an application is in testing phase, it is difficult to go back and change as the steps are documented before completion of the step.

### **Modified Waterfall Model:**

The main difference between pure waterfall model and modified waterfall model is the feedback system which provides to previous phases. This model works better when the requirements are well understood. It allows developers to return to a previous phase for verification or validation, confining to connected steps. The modified waterfall model uses the same phases as of pure waterfall model. This enables the phases to overlap when needed. The modified waterfall can also split into subproject at an appropriate phase.



We have chosen Modified Waterfall Model because of the feedback system. While developing phase, if we aren't clear about the requirements or if the requirements couldn't be implemented, this model provides the flexibility to go back to the previous steps and go through the requirements. If there is continuity between phases, then documentation can be reduced. While in testing phase, if there are any errors in the software, they can be rectified by moving back to the development phase. Our product will be developed in Frontend and Backend, where frontend will be user interface and backend will be the implementation, therefore Modified Waterfall Method will allow the developers to perform more rigorous checking for this kind of project where both the operations can be monitored, altered and be updated as necessary.

## PROJECT DEFINITION

This product can be used and will be helpful to people who need to communicate with one other without providing their personal information. They can use this product to text and message one another for specific needs, for example a group of students in a class working on a small one-day project where they need to communicate can easily join the chat room. They don't need to worry about privacy or the chat session as it will be deleted once they leave the session. Another use case can be a group of strangers who met at an event or at library and found to be sharing same interest on a technology or a topic or a book, can text one another, share their views without giving up their mobile number or E-mail ids. The users will be provided with an interactive chat interface where they can enter multiple line text messages in a text box and can use emojis and send it. The receiver will get the message instantly and can reply. The chat heads will be the names that can be given by the users themselves at the beginning of the chat session before they join the chat box.

## PROJECT ORGANIZATION

Work Breakdown structure of the project with respective schedule of team's task.

### Backend Development:

Creating socket for message transfer from client to server and server to client.	02/20/2021
Integrating Emojify.	03/06/2021
Testing the development.	03/20/2021

### Frontend Development:

Creating Web Pages.	04/03/2021
<ul style="list-style-type: none"><li>• User entry page</li><li>• Authenticating Room Key</li><li>• Creating Text Box</li></ul>	
Integrating Frontend and Backend.	04/17/2021
Final Testing.	05/01/2021

## VALIDATION PLAN

Firstly, the user will be prompted to enter their name of choice.



A rectangular form with a dark border. At the top, the text "Enter Your Name" is centered. Below it is a horizontal text input field. At the bottom right, the text "Next" is displayed.

Enter Your Name

Next

After user enters his name, they will be asked the chat room key they want to enter.



A rectangular form with a dark border. At the top, the text "Enter Chat Room Key" is centered. Below it is a horizontal text input field. At the bottom right, the text "Enter" is displayed.

Enter Chat Room Key

Enter

Finally, the user can enter the chat room and text to other users.

Chat Room

User 1

Text....

User 2

Text...

Enter your message 🤔

## CONFIGURATION/ VERSION CONTROL

This project will be implemented and deployed with all the specifications mentioned, without any incremental additions after deployment, therefore no Revision Control Software is being used for this project.

## TOOLS

The Software and Hardware tools required for this project are:

A computer with i5 or greater generation processor.

4GB or more RAM

512 GB of storage.

Windows 10 or Mac OS.

Microsoft Visual Studio.

Notepad++

The technologies used are: ReactJS, NodeJS, Express, CSS, Emojify