

# Project 1: Online Chat Application “QuickChat”

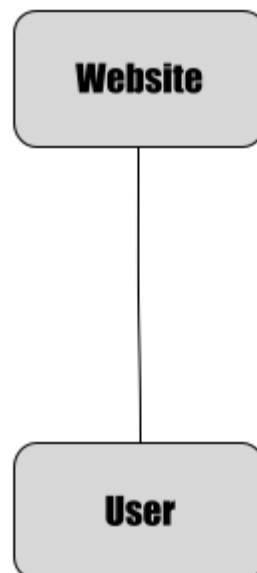
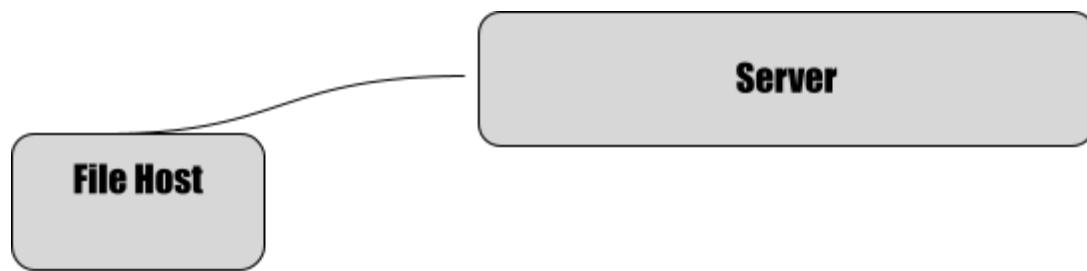
## Product Vision

**For** individuals and organizations that need fast and secure communication across online devices, **the** “QuickChat” app is a real-time chat application **that** enables users to message between each other in real time in both public and private chat rooms, private messaging, and in creating profiles with a username and avatar. **Unlike** other traditional messaging app competitors, **our product** offers an easy to use interface, with accessibility for both personal and business use.

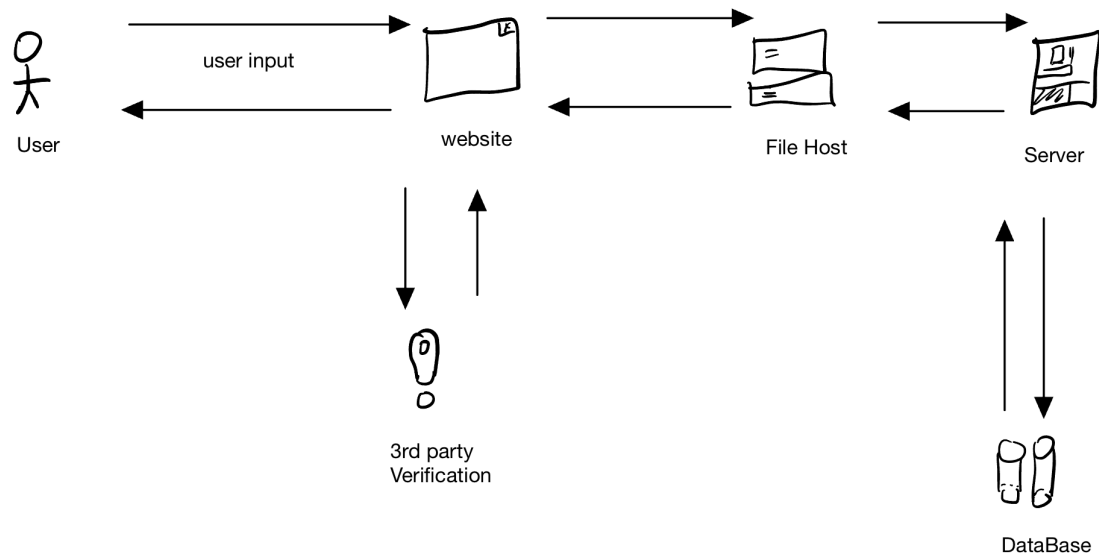
<p>Functional requirements (How the system must work)</p> <p>Business Requirements</p> <ul style="list-style-type: none"><li>- The application must scale to accommodate a growing user base. Including supporting 1 million active users by the first year, and must scale to accommodate any additional users.</li><li>- A free-to-use service may be accepted for rapid customer growth, but the application must generate profits by the 5th year.<ul style="list-style-type: none"><li>- The chat application must result in long-term profit for the business.</li></ul></li></ul>	<p>Nonfunctional requirements (how the system should operate, focusing on quality attributes)</p> <p>Business Requirements</p> <ul style="list-style-type: none"><li>- A buy-out of the company (for its customer data) may be a way to monetize.</li><li>- Selling customer data directly to advertisers may also be a way to monetize.</li><li>- Banner ads and paid services (for unlimited images, or personalized user profiles) can provide scaling revenue.</li></ul>
<p>User Requirements</p> <ul style="list-style-type: none"><li>- Users can register in the app with their email, username, and password.</li><li>- Registered users log into the system with their ID and password.</li><li>- Users can message each other in near-real time.</li><li>- Users can join public and private chat rooms</li><li>- Users can send images and files within a group chat or private</li><li>- Users must be over the age of 13 to</li></ul>	<p>User Requirements</p> <ul style="list-style-type: none"><li>- User information must be stored in an encrypted and redundant database.</li><li>- Email verification or phone verification can be used to deter spam accounts.</li></ul>

register for an account.conversation.	
<p>System Requirements</p> <ul style="list-style-type: none"> <li>- The system must have a website.</li> <li>- The system must place a limit on file and image sizes and quantity (e.g. a max transfer of 5 megabytes per 10 minutes)</li> </ul>	<p>System Requirements</p> <ul style="list-style-type: none"> <li>- The application must be compatible with web browsers like Chrome.</li> <li>- The system must deliver the messages with a latency of under 5 seconds, even during peak usage.</li> <li>- The system must be user friendly, and have clear buttons for clicks and tasks.</li> </ul>

### **Context Diagram**



Context Diagram



**To do:**

- Add the Project/Product Vision
- Add 5+ project requirements (both Functional and Non-Functional) in use case & use story format, (refer to format from lecture notes).
- Add a context diagram with all actors and details that will depict the entire system.

**Submit:**

- A .zip file with our project repository files (with Readme.md file)
- Link to our Github repository (with Readme.md file)