Software Requirements Specification

for

GiGe

Prepared by

Tanmay Bansal(19BCE0421)

Atul Agarwal(19BCE0436)

Aaditya Pareek(19BCE0866)

24th March, 2021

Table of Contents

Table of Contents

1. Introduction

1.1 Purpose

The Purpose of the product is to provide university students with an interface that allows them to share their products amongst themselves and earn rewards in this process. The product is mainly aimed at reducing expenses for students as well as earning out of their excess or unutilized resources.

Students often find themselves in the need of daily essentials and that too at an affordable price. Also, there are some students that have too much stuff that they don't even use now. GiGe provides a quick and simple solution to this problem by enabling the users to share their goods in exchange for Gicoins.

1.2 Document Conventions

The document follows the IEEE template and format. The headings are bold with Headings 2 size with a Times New Roman font. The paragraph content is written with a normal text style and Times font, its size is set to 18. Bold highlighting has been used to emphasize section and subsection headings. All the content regarding subheadings are in list and paragraph form.

1.3 Intended Audience and Reading Suggestions

The SRS is intended for different types of readers such as:

- Developers
- Designers
- Project Managers
- Product Managers
- Marketing Managers
- Users
- Copywriters

The rest of SRS contains
Introduction that includes Purpose, Document Conventions, Intended Audience, and Reading and Suggestions and Product Scope and References

Overall Description which includes Product Perspective, Product Functions
User Classes and Characteristics, Operating Environment, Design and Implementation Constraint, User Documentation and Assumptions and Dependencies

External Interface Requirements which includes User Interfaces, Hardware Interfaces, Software Interfaces, and Communications Interfaces.

List of all System Features that are important

Other Nonfunctional Requirements include Performance Requirements, Safety Requirements, Security Requirements, Software Quality Attributes, and Business Rules.

Other Requirements are related to the different users and readers of the document.

1.4 **Product Scope**

GiGe is a platform for sharing goods and services among students using a Coin system. Gicoins can be purchased as well as earned. Users can upload "gives" and "gets" which are the way of changing points. In "get", you spend points by getting some items, and in "give", you earn points by giving your items to students. That item can be almost anything like books, video games, electronics, projects, cycles, sports goods, etc. The giver has to upload details about the service.
It is basically an ecosystem that thrives on people helping each other. The coins are used to measure how much one is helping others. Students who "give" earn coins and can spend those coins in the app to "get" things they need.

1.5 References

Fiverr is a freelance website for all kinds of remote work. It has two modes- buyer and seller- similar to our platform. So we have taken some references from this website.
Fiverr website :https://www.fiverr.com

myTurn combines the best and most essential features of asset tracking, rental, and product subscription services.
Myturn: https://myturn.com

Rentbaaz is a startup that was started by VIT students to rent things on campus and it has now grown into a B2B (business to business) rental company but it started as a peer to peer rental similar to our idea - GiGe.
Website : https://rentbaaz.com

2. Overall Description

2.1 Product Perspective

GiGe is an online marketplace for university students. It allows users to earn out of their extra resources by renting out the products for short durations of time. This project is completely new as the functionalities and the website being developed on the particular idea is an innovative one.

Users will have two roles on this site, GIVE and GET.

By default, each user is assigned the role of GET, however, they can choose to have both roles. A GIVE user will upload their products for rent on the site, the GET user will be the rentee in this case. The currency of exchange here will be Gicoins.

The users on login will see the GET mode wherein the whole marketplace of the site is displayed. The user can choose and add any product of their liking to their carts. On checkout, the transactions will take place.

A user could alternatively choose to enter the GIVE mode. Here they can upload their products to the marketplace for rent. If their particular product is rented out, they earn Gicoins.

All other functionalities of the site are built to improve the experience and make the user's work easier.

2.2 Product Functions

GiGe is intended to have two user modes, Give and Get. In Give, the user can upload their products on the site for rent, and try to earn out of these. In Get, a user can check if any item in the site marketplace is of their particular need or interest.

Main Features of the Site:

1. Login/Logout/Account creation
2. Product Sell/Give Mode
3. Product Buy/Get Mode
4. User Reviews and Ratings for Products and services
5. Searching for products and services
6. Transaction Maintenance(Gicoin transactions made safe by Cryptography)
7. Account View
   a. Wallet : GiCoins left and spent
   b. Product Selling History
   c. Product Purchase History

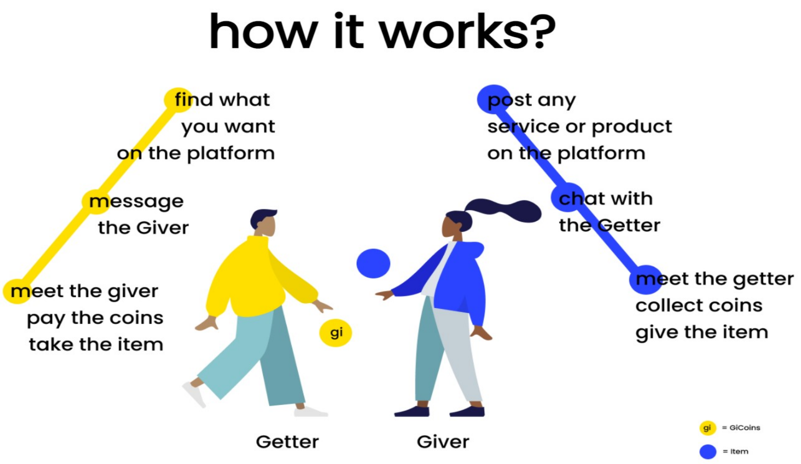
2.3 User Classes and Characteristics

The various types of users of our platform include:

Giver:
The person who uploads the details of the item or service so that others on the platform can buy or rent that for some Gicoins. This user acts as a seller of a particular service or good to others in the GiGe ecosystem. For renting or selling to others the Giver receives some benefits such as Gicoins which can be used later when the user acts as a Getter in case of some other product.

Getter:
The person who browses and searches our platform to get something they want to rent or buy from a Giver. To get that service or product the Getter has to use Gicoins. They connect with the Giver through our platform and then the giver and getter can meet to exchange the goods and Gicoins.

College Students :
College students are main users as they can easily share goods and services with their friends and others on their college campus. Also, they don't have a lot of money that can be spent on buying new things so our sharing platform can help them in getting things they require using Gicoins.


how it works?
find what you want on the platform
message the Giver
meet the giver pay the coins take the item
post any service or product on the platform
chat with the Getter
meet the getter collect coins give the item
gi
Getter
Giver
gi = GiCoins
= Item

2.4 Operating Environment

a.Hardware Requirements:

Hardware requirements are expected to be minimal. The most important part will be a fast hard drive for the MongoDB Database(PostSQL) to reside on and optimal network speed. The client side of the application is very portable since the only required piece of software is a W3c Compliant Web Browser. Microsoft Windows, *Nix, and Mac Os all have browsers that will work with the application. Following are the minimum specifications for the various computers and other hardware components:

I. Server Requirements:

a. Minimum CPU - 1.9 gigahertz (GHz) x86- or x64-bit dual core processor with SSE2 instruction set
b. Minimum Disk Space: 2GB RAM
c. Minimum Memory: 512 MB
d. Bandwidth greater than 50 KBps (400 kbps)
e. Latency Under 150ms
f. Required Software - Django, MongoBD, HTML, CSS, Javascript, PostSQL

II. Client Requirements:

a. Minimum CPU: 1.9 gigahertz (GHz) x86- or x64-bit dual core processor with SSE2 instruction set
b. Minimum Disk Space: 512 MB
c. Minimum Memory: 256 MB
d. Bandwidth greater than 50 KBps (400 kbps)
e. Minimum Display: Super VGA with a resolution of 1024 x 768
f. Required Software: W3C Compliant Web Browser

b. Software Requirements:

There are no explicit requirements for client side users(except Internet Browser) as the product is to be implemented as a website.

However the development team throughout the course of development the software technologies used are:

1. Project Libre
2. MS Project
3. Online Visual Paradigm
4. projectmanager.com
5. Visual Studio
6. Figma

2.5 Design and Implementation Constraint

The challenges regarding this product will be to maintain a transaction ledger for users and to develop a time and space-efficient search algorithm.
On implementation, an internet connection will be required to access and use the features of the site. The memory space on the server will be limited, so storage memory cannot exceed the value.

2.6 User Documentation

Along with the software product, a user tutorial video would be created to help people understand the working methodology and usage of the developed system. It would be created for nontechnical individuals and the level of content or terminology would be basic and concepts would be explained in layman's terms. The user tutorial video would capture the purpose and scope of the product along with key system features and operations; step-by-step instructions for using the system including conventions, messaging structures, quick references, tips for errors and malfunctions.

2.7 Assumptions and Dependencies

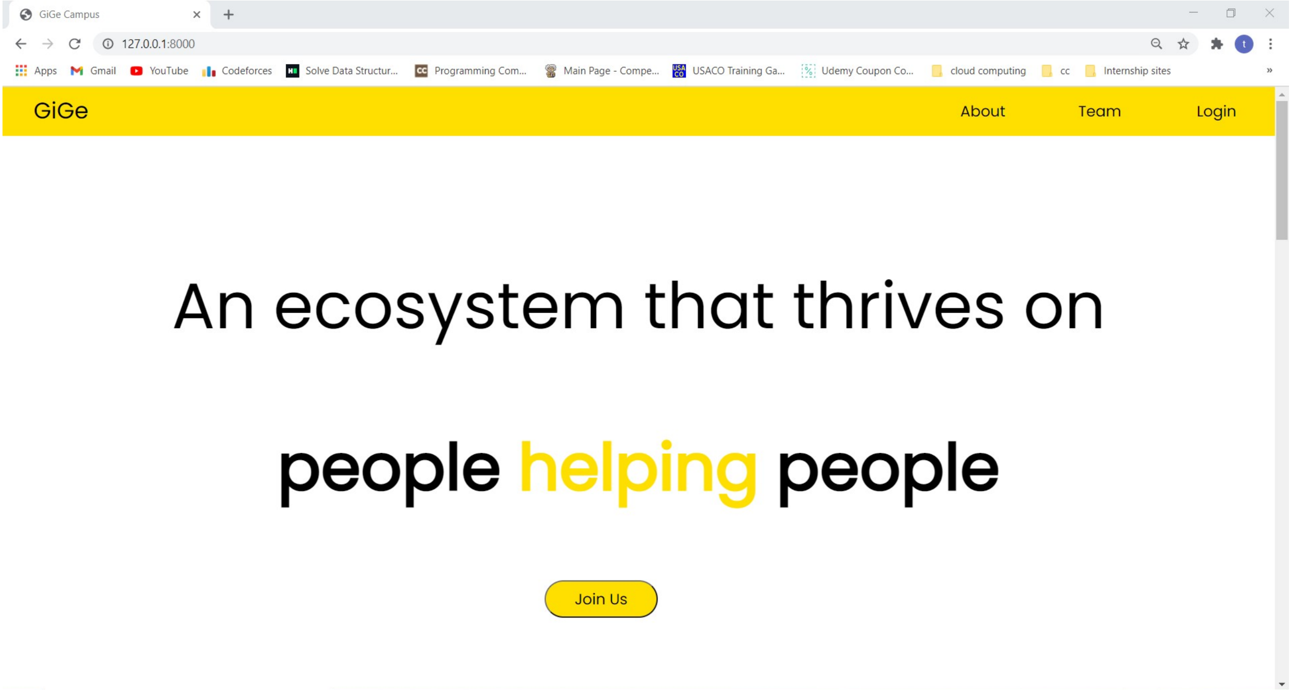
1. It is assumed that the quality of service provided by a hosting service provider is optimal and up to the mark.
2. It is also assumed that any user of the site is comfortable with online systems and can navigate through a web interface with some level of comfort and ease.
3. GiGe also assumes that it will always have a product for rent database entity and hence it's supply will never run dry.

3. External Interface Requirements

3.1 User Interfaces

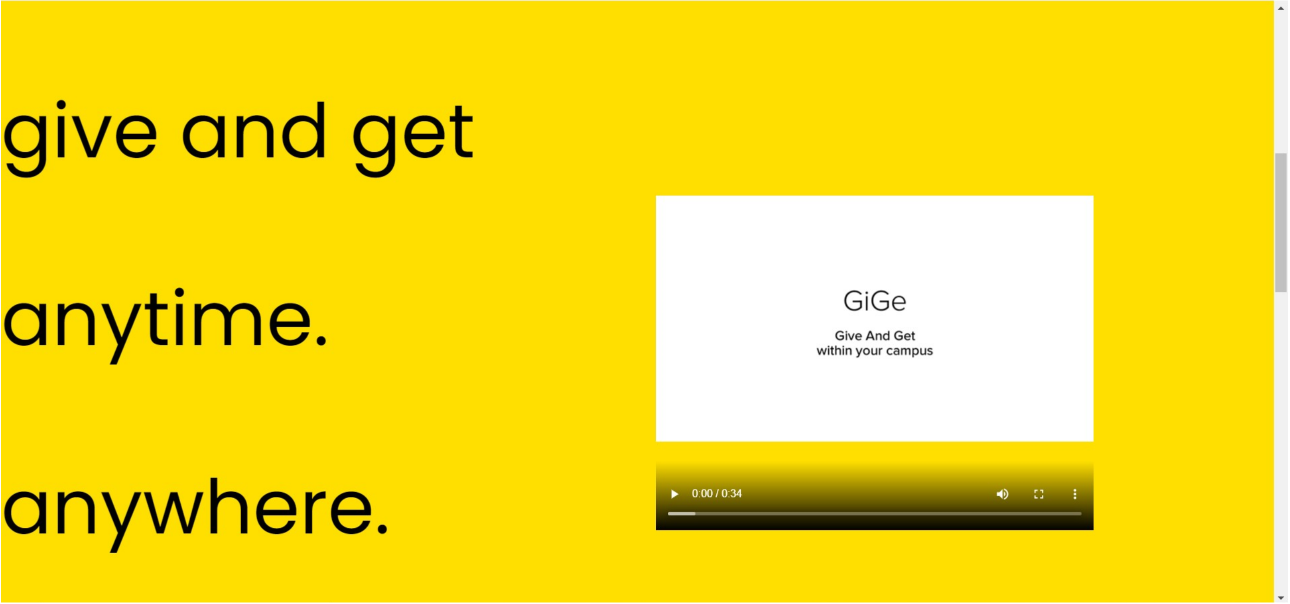
This section describes the logical characteristics of each interface between the intended software product and the users. For user interface design, common GUI standards will be followed along with the presence of keyboard shortcuts, error message display standards etc., and standard buttons and functions (i.e. help) will appear on every screen. Details of the user interface design are intended to be documented in a separate user interface specification.
As for the various interfaces, we will have a home interface, login interface, signup interface, give mode interface, and the get mode interface. Till now we have completed our home interface, login interface, and sign up interface.

Home page



An ecosystem that thrives on

people helping people



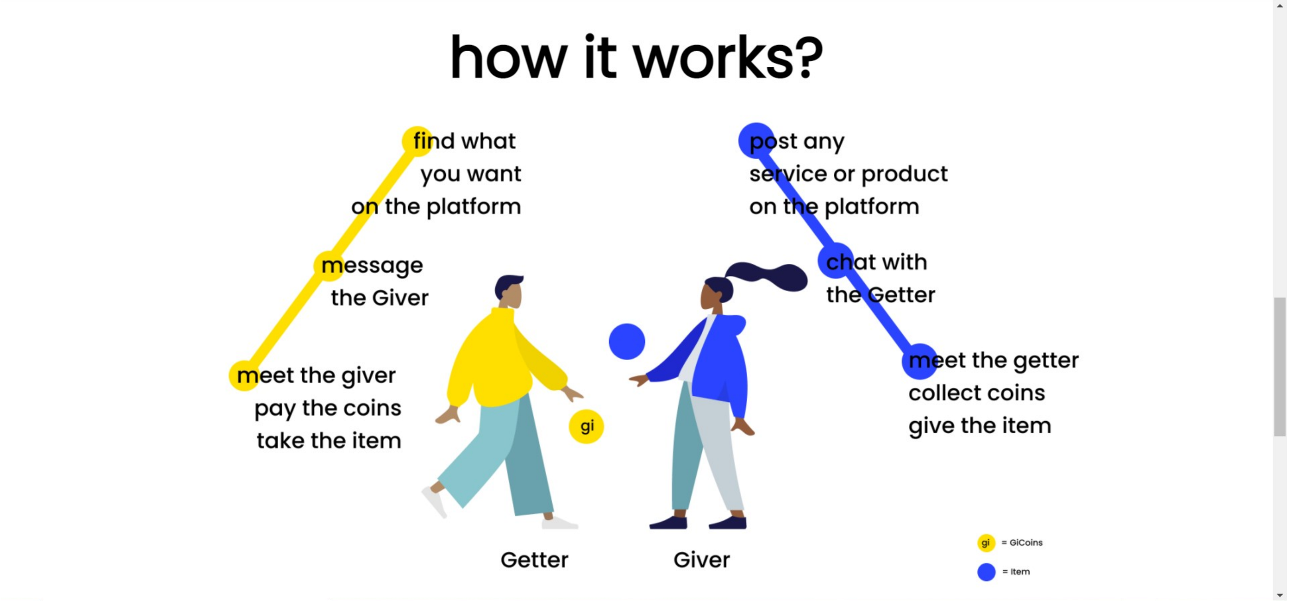
give and get

anytime.

anywhere.

GiGe

Give And Get
within your campus



how it works?

find what
you want
on the platform

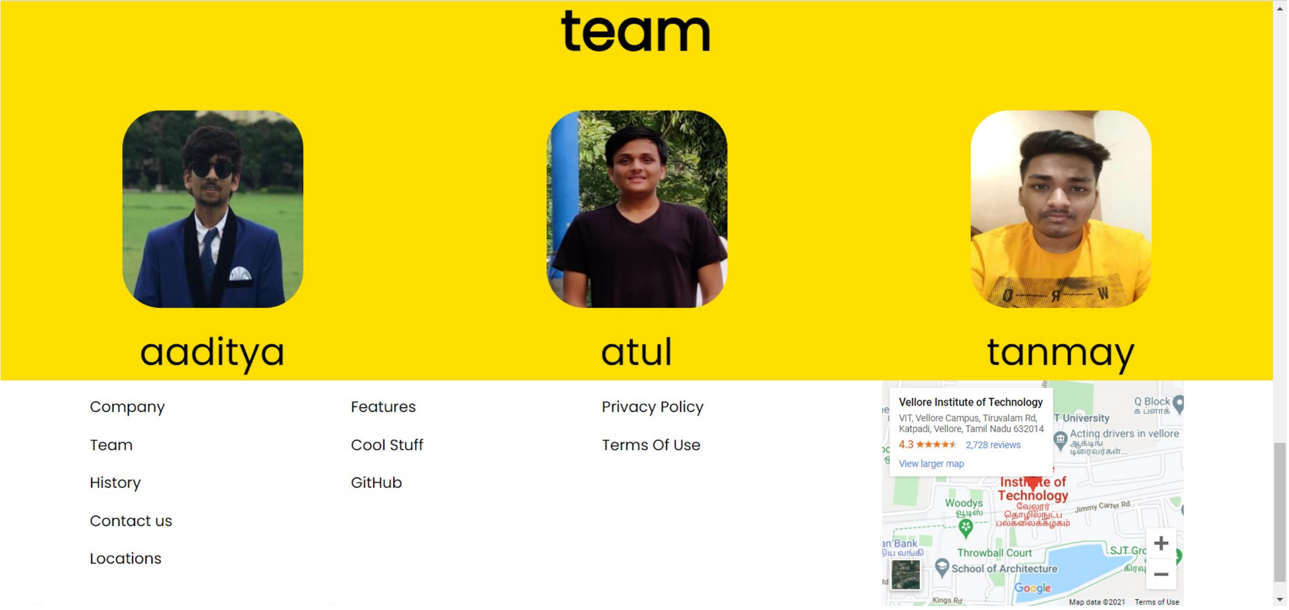
message
the Giver

meet the giver
pay the coins
take the item

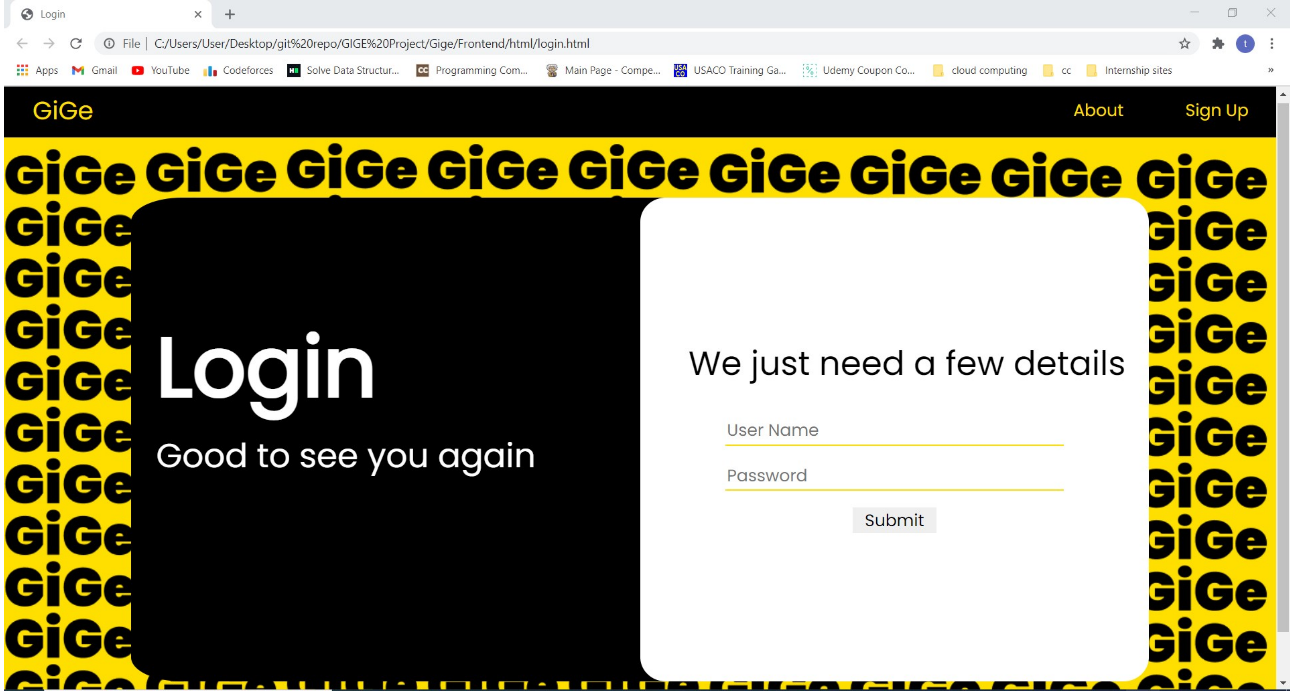
post any
service or product
on the platform

chat with
the Getter

meet the getter
collect coins
give the item



Login page



Login

Good to see you again

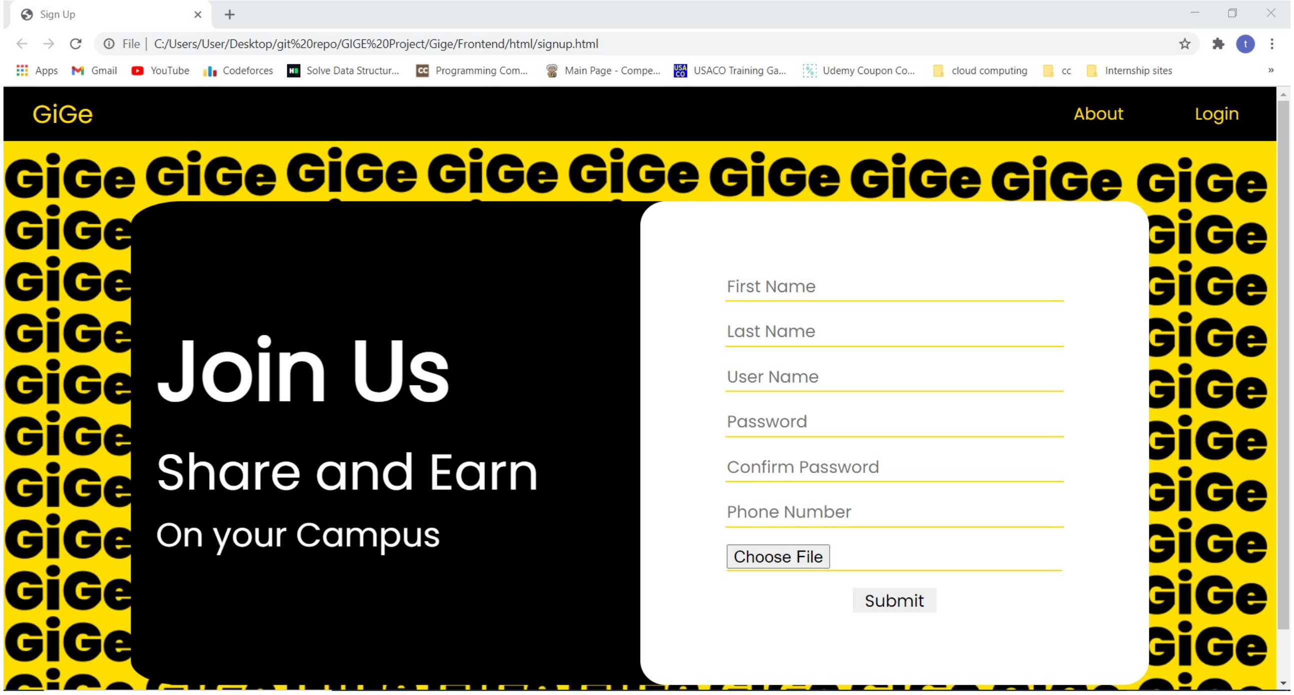
We just need a few details

User Name

Password

Submit

Signup page



3.2 Hardware Interfaces

There are three external hardware devices used by the shopping portal, each related to a user interface. These devices are surface computers, wireless tablets, and touch displays. All three devices must be physically robust and immune to liquid damage and dust. The devices behave as 'terminals' in the sense that they never have a full system image, do not store data, and are not used for the core logic of the system.

Devices include:

a. Django server with high performance and use for CPU usage.

b. Database server using Postgresql with increased use of memory space.

c. Computers with Windows and Linux users will work.

However, they should be fully capable computers that can use textual data from the server along with local UI/interpretation code to display UI elements and take input. All order and transaction records should be stored on the server, not these computers.

3.3 Software Interfaces

The GiGe website will interface with a Database Management System (DBMS) that stores the information necessary for the website to operate. The DBMS must be able to provide, on request and with low latency, data concerning the item catalogue, customers (and their passwords) and available orders. Additionally, it should take and archive data provided to it. This data will include records of all orders and transactions (system states and state changes) made by the user. The DBMS must store all Data such that it can be used for accounting, as well as accountability.For the prototype, the software requirements are to support windows operating systems with support to PostgreSQL and Django servers.

3.4 Communications Interfaces

The website will interface with a Local Area Network (LAN) to maintain communication with all its devices. It should use a reliable-type IP protocol such as TCP/IP or reliable-UDP/IP for maximum compatibility and stability. An Internet connection and a web browser are required in order to make use of several functions and to be executed such as searching, viewing and downloading. All devices it will interface with should contain standard Ethernet compatible, software accessible LAN cards to maintain communication between the server and the surface computers, tablets, displays and the external payment system.

4. System Features

The major services and functional requirements for the product can be illustrated by system features.

4.1 Login/Signup

4.1.1 Description and Priority

A login/signup feature is one of the most essential and important features for a user based website. It allows users to create sessions and makes it possible to identify users and store or retrieve their information from the database. It is one of the most important features of any site and it's functioning and execution have to be smooth.

4.1.2 Stimulus/Response Sequences:

1. User clicks on Login Link on Home Page
2. Site Connects to Login Page, Requests Uses to fill input fields, Username, and Password
3. User Fills the form and Clicks on Submit Button
4. Site Connects to DATABASE and takes user's input values and checks for a match in the database
5. If found, Success, move to Marketplace page
6. Else, Display Invalid input, ask to fill the form again, or Sign Up for the site.

4.1.3 Functional Requirements

REQ-1: The interaction protocol between Web Server-Client, Client-Server is defined.

REQ-2: User Details Table Created in Database

4.2 Product Sell Mode:

4.2.1 Description and Priority:

One of the main features of the site. A Sell mode will allow users to add their products for sale on our site or check the current status of their existing products. It is a high priority feature of the site. The feature allows users to earn Gicoins on the site.

4.2.2 Stimulus-Response Sequence:

1. User Clicks on Give Button in the Navigation Bar.
2. Site Sends Request containing user token id to the server.
3. The server responds with appropriate values from the database.
4. Site Displays list of all uploaded products by the user, a button exists on the top right to list new products on the marketplace.
5. User clicks on upload new product, the server responds with a form for product details.
6. The user fills the form and clicks on the submit button.
7. Form data is sent to the server and values are updated in the product database.

4.2.3 Functional Requirements:

REQ 1: The interaction protocol between Web Server-Client, Client-Server is defined.

REQ 2: Product Details Table in Database.

REQ 3: Converting Data tokens from frontend to backend query language.

4.3 Product GET Mode:

4.3.1 Description and Priority:

The other end to product upload feature. A GET mode allows users to view the marketplace of the site. It is the first page that loads on successful login/sign up. The products are listed used CSS Grid and Box Models to make it easy for users to navigate and get a better understanding of the uploaded product.

4.3.2 Stimulus/Response Sequences:

1. Site Displays all listed products after successful user login/sign up.
2. User clicks on add to cart option for any product after viewing details.

3. Server stores all details in a javascript object.
4. The user clicks on the cart option.
5. The server receives requests and passes tokens to the transaction page.

4.3.3 Functional Requirements:

REQ 1: The interaction protocol between Web Server-Client, Client-Server is defined.
REQ 2: Product Details Table in Database.
REQ 3: Converting Data tokens from backend to frontend design.

4.4 Product Review:

4.4.1 Description and Priority:

After the completion of the time duration, both the users will get a form to place a rating on their counterpart. This feature is of low priority and will be optional.

4.4.2 Stimulus/Response Sequences:

1. After completion of the rent cycle, the site displays a rating form for both parties.
2. They can choose to click on any of the 5-star rating options.
3. Their response is sent to the server.
4. The average rating of any user is determined on the basis of these ratings.

4.2.3 Functional Requirements:

REQ 1: The interaction protocol between Web Server-Client, Client-Server is defined.
REQ 2: Review Attribute in product relation.
REQ 3: Time check function in the backend.

4.5 Product Search:

4.5.1 Description and Priority:

In the GET mode, the user is given a marketplace site. Scrolling and looking for a specific product will be tedious and inefficient. Hence a search feature for the page becomes important and necessary.

4.5.2 Stimulus/Response Cycle:

1. A search input form will be present in the navigation bar.
2. The user enters the keyword and submits the form.
3. The values are sent to the server and run against keywords in the database.
4. If matches are found, data is sent back to the client-side and only the keyword accepted products are displayed.

4.5.3 Functional Requirements:

REQ 1: The interaction protocol between Web Server-Client, Client-Server is defined.

REQ 2: Keyword Attribute in product relation.
REQ 3: Efficient searching algorithm.

4.6 Product Transaction:

4.6.1 Description and Priority:

After any user in GET mode decides to temporarily purchase the products, Gicoins have to be transferred from their account to the sellers account. This feature will be important to complete the site's purpose.

4.6.2 Stimulus and Response:

1. In the GET mode cart, user clicks on the purchase option.
2. This request along with buyer, seller and product tokens are sent to the server.
3. Server checks the database if all values are above limit and valid.
4. Purchase is verified.
5. Both parties are sent notifications from the server.

4.6.3 Functional Requirements:

REQ 1: The interaction protocol between Web Server-Client, Client-Server is defined.
REQ 2: Transaction Relation.
REQ 3: Ledger creation and maintenance algorithm.

4.7 Account View:

4.7.1 Description and Priority:

Users will be able to view their transaction and purchase history. It is not a high priority feature.

4.7.2 Stimulus and Response:

1. In the navigation bar dropdown, user can click on any the three options:
   a. Wallet : GiCoins left and spent
   b. Product Selling History
   c. Product Purchase History

2. Requests are sent to the server and it responds with appropriate values from the database.
3. The received values are displayed on the page.

4.7.3 Functional Requirements:
REQ 1: The interaction protocol between Web Server-Client, Client-Server is defined.
REQ 2: Converting Data tokens from backend to frontend design.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

In order to maintain an acceptable speed at a maximum number of uploads allowed from a particular customer as any number of users can access the system at any time. Also, the connections to the server will be based on the attributes of the user like his location and the server should be available around the clock.

5.2 Safety Requirements

The passwords of the users should be hashed. Also other sensitive data cannot be leaked, so data safety protocols become necessary. Routine backups of existing databases to ensure data integrity and no loss in case of system failure.

5.3 Security Requirements

As it will be an online system, the database can be exposed to injection attacks, protection against these becomes a must and of utmost importance.

5.4 Software Quality Attributes

Reliability :
The solution should provide reliability to the user that the product will run with all the features mentioned in this document are available and executing perfectly. It should be tested and debugged completely. All exceptions should be well handled.

Accuracy :
The solution should be able to reach the desired level of accuracy. But also keeping in mind that this prototype version is for proving the concept of the project.

Security:

Security is a priority when a software is tested especially when it is built in such a way that it contains some crucial information when leaked can cause harm to business. The system will have the ability to protect data and defend information from unauthorized access. Security also includes authorization and authentication techniques, protection against network attacks, data encryption, and other risks.

5.5 Business Rules

Whenever a new account is made we give them a certain number of coins so they can start using GiGe after that they have to contribute to the GiGe ecosystem to earn coins and then spend those. GiGe is an ecosystem that thrives on people helping each other. The coins are used to measure how much one is helping others. Students who "give" earn coins and they can spend those coins in the app to "get" things they need.