**Quick PC**

*PC Building Assistance Mobile Application*

****

**Test Specification Documentation**

CECS 491A T/TH 12:30 PM

# **Table of Contents**

[**Table of Contents**](#_id9gyep3b8ui) **1**

[**Abstract**](#_ojmnhpk0q2eo) **2**

[**Unit Level**](#_ak4b52a56izx) **3**

[**Module Level**](#_vhzpuw9lv08d) **6**

[**Integration Level**](#_kq9jorhp37ce) **9**

[**System Level**](#_m34ewqq7d0vi) **12**

[**Acceptance Level**](#_sb29w08rdpju) **17**

# 

# 

# 

# 

# 

# 

# **Abstract**

This document will consist of five levels of testing (Unit, Module Integration, System, or Acceptance). The testing will be done to maintain data integrity, ensure usability, functionality. Each test case will consist of the test level being used, quality criterion, description of test, steps of the test case, and expected outcome. A brief description of each point is listed below.

1. **Test Level:** The level being tested (Unit, Module, Integration, System, or Acceptance)
2. **Quality Criterion:** The element that is being tested
3. **Description of Test:** A brief description of what is being tested
4. **Steps of the Test Case:** A step by step explanation of how the test was done
5. **Expected Outcome:** The expected outcome of the test that have been completed

# 

# 

# 

# 

# 

# 

# 

# 

# 

# **Unit Level**

For our Unit Level testing, we wanted to ensure that the individual units or components of our QuickPC application functioned correctly. We tested individual parts for sign up, login, search for parts, build pc parts compatibility, and view a pc information page to ensure its effectiveness with other Unit components. The intention of this level to ensure that the software runs appropriately and effectively.

1. **Sign Up**

|  |  |
| --- | --- |
| Test Level | Unit |
| Quality Criterion | Functionality |
| Description of Test | Users are able to register an account and access the homepage with the new account. |
| Requirements References | 1. (Use Case #2) Creating an Account 2. (Use Case #3) Login 3. (Use Case #4) View Home Menu |
| Steps of the Test Case | 1. Navigate to the sign up page 2. Enter details that are required to sign up    1. Unique Email Address    2. Unique 6 character password 3. Press sign up |
| Expected Outcome | Users are able to create an account which is stored into the database. |

1. **Login**

|  |  |
| --- | --- |
| Test Level | Unit |
| Quality Criterion | Functionality |
| Description of Test | Users are only able to login using their existing account and are able to access the homepage. |
| Requirements References | 1. (Use Case #2) Creating an Account 2. (Use Case #3) Login 3. (Use Case #4) View Home Menu |
| Steps of the Test Case | 1. Navigate to the login page 2. Enter details that are required to login    1. Email    2. Password 3. Press login |
| Expected Outcome | Users are able to access the homepage after logging in using their account credentials. |

1. **Home Page: Search for Parts**

|  |  |
| --- | --- |
| Test Level | Unit |
| Quality Criterion | Efficiency |
| Description of Test | User is able to search for a specific pc component part in a quickly manner |
| Requirements Reference | 1. (Use Case #5) Search for parts with prices, features, etc 2. (Use Case #17) View a PC Information Page |
| Steps of the Test Case | 1. Navigate to the homepage 2. Select “Search Parts” 3. Select a type of component 4. Verify the list of components |
| Expected Outcome | Users are able search for a specific part in the database. |

1. **Home Page: Build PC Parts Compatibility**

|  |  |
| --- | --- |
| Test Level | Unit |
| Quality Criterion | Usability |
| Description of Test | System is able to check if the parts are compatible with each other. |
| Requirements Reference | 1. (Use Case #5) Search for parts with prices, features, etc 2. (Use Case #6) Managing budget of list parts |
| Steps of the Test Case | 1. Navigate to the homepage 2. Select “Build PC” 3. Select a type of part 4. Select “Search” 5. Verify if part compatibility is accurate |
| Expected Outcome | Part is not added into the build PC list if part is not compatible with the current build. |

1. **View a PC Information Page**

|  |  |
| --- | --- |
| Test Level | Unit |
| Quality Criterion | Usability |
| Description of Test | Users are able to access the PC information page after selecting a specific computer part. |
| Requirements Reference | 1. (Use Case #17) View a PC Information Page |
| Steps of the Test Case | 1. Navigate to the homepage 2. Select “Search Parts” 3. Select a component 4. Select a specific part 5. View the specifications |
| Expected Outcome | User is able to see the specifications of the specified computer part. |

# **Module Level**

For the Module level we test the major functions of Quick PC. We first examine creating and using accounts, then searching, sorting, and viewing parts, then finally selecting parts for a new pc build. The goal is to ensure that different units of the code work together for major functions.

1. **User Registration & Login**

|  |  |
| --- | --- |
| Test Level | Module |
| Quality Criterion | Functionality |
| Description of Test | Users are able to sign up and then login using the newly created account. |
| Requirements Reference | 1. (Use Case #2) Creating an Account 2. (Use Case #3) Login 3. (Use Case #4) View Home Menu 4. (Use Case #7) Database Storage: User Login Information |
| Steps of the Test Case | 1. Navigate to the sign up page 2. Enter details that are required to sign up    1. Unique Email Address    2. Unique 6 character password 3. Press sign up 4. Enter details that are required to login    1. Email    2. Password 5. Press login |
| Expected Outcome | Users are able to access the homepage after logging in with their newly created account. |

1. **Account Settings**

|  |  |
| --- | --- |
| Test Level | Module |
| Quality Criterion | Usability, Reliability |
| Description of Test | Verify whether a user can view and change the information on their account through Account settings. |
| Requirements Reference | 1. (Use Case #2) Creating an Account 2. (Use Case #3) Login 3. (Use Case #7) Database Storage: User Login Information 4. (Use Case #9) Product Alert Sign up |
| Steps of the Test Case | 1. Log in to account 2. Select Account from the home page 3. Select change username, password, or email 4. Enter changes for each 5. Click save changes 6. Log out 7. Log back in 8. Select Account and verify changes |
| Expected Outcome | The changes the user made will be stored and pulled from the database successfully and displayed. |

1. **Search for Parts**

|  |  |
| --- | --- |
| Test Level | Module |
| Quality Criterion | Usability |
| Description of Test | Users can search for a type of part and are able to filter or compare the parts in a list generated from the database. |
| Requirements Reference | 1. (Use Case #5) Search for parts with prices, features, etc 2. (Use Case #8) Database Storage: PC Component Info 3. (Use Case #14) View Component Comparison |
| Steps of the Test Case | 1. Navigate to the homepage 2. Select “Search Parts” 3. Select a component 4. Select the filter tab and select filters 5. View filtered list 6. Select specific parts 7. Select the compare tab |
| Expected Outcome | The user is able to view a narrowed down list of parts. |

1. **Part Info Page**

|  |  |
| --- | --- |
| Test Level | Module |
| Quality Criterion | Usability |
| Description of Test | A product selected from the search list will have its specific data, prices, and reviews pulled from the database which is then presented to the user. |
| Requirements Reference | 1. (Use Case #5) Search for parts with prices, features, etc 2. (Use Case #17) View a PC Information Page 3. (Use Case #8) Database Storage: PC Component Info 4. (Use Case #13) View Component Reviews 5. (Use Case #15) Graphs for Price History |
| Steps of the Test Case | 1. Select a specific part from the search list 2. Scroll down the info list 3. Select the Prices tab 4. Scroll down the list of stores and their prices 5. Select the Reviews tab 6. Scroll down the list of reviews. |
| Expected Outcome | The user can see all the info related to a specific part. |

1. **Build PC Page**

|  |  |
| --- | --- |
| Test Level | Module |
| Quality Criterion | Usability, Functionality |
| Description of Test | The user will search and select the parts needed to build a new pc. |
| Requirements Reference | 1. (Use Case #5) Search for parts with prices, features, etc 2. (Use Case #6) Managing budget of list parts 3. (Use Case #8) Database Storage: PC Component Info |
| Steps of the Test Case | 1. Select Build PC from the home page 2. Select a type of part from the build list 3. Select a specific part from the search list 4. Repeat steps 2 & 3 for different parts until all the parts needed have been selected. |
| Expected Outcome | The user is presented a list of compatible parts that form a complete PC. |

# **Integration Level**

For the integration level testing we assessed the interconnected subsystems utilized in Quick PC. These subsystems include interactions with third party applications such as API’s. Cross platform compatibility between IOS and Android is also examined for quality and reliability. The main objective of integration testing is to verify that associated modules are collaborating correctly.

1. **Firebase User Authentication**

|  |  |
| --- | --- |
| Test Level | Integration |
| Quality Criterion | Functionality & Reliability |
| Description of Test | The user will create an account and login with verified credentials |
| Requirements Reference | 1. (Use Case #2) Create an account 2. (Use Case #3) Login 3. (Use Case #7) Database Storage: User Login Information |
| Steps of the Test Case | 1. Navigate to the sign up page 2. Enter user account details 3. Sign up 4. Navigate to the login page and enter generated credentials 5. Login |
| Expected Outcome | The user account is verified by the Firebase authenticator resulting in a successful login. |

1. **Firestore Database**

|  |  |
| --- | --- |
| Test Level | Integration |
| Quality Criterion | Efficiency |
| Description of Test | Checks that the Quick PC application is retrieving live information from the cloud Firestore Database |
| Requirements Reference | 1. (Use Case #5) Search for parts with prices, features, etc 2. (Use Case #8) Cloud Database PC Component Information 3. (Use Case #17) View a PC Information Page |
| Steps of the Test Case | 1. Navigate the search for parts page 2. Enter a specific component 3. Verify that the database is returning the components information |
| Expected Outcome | Most up to date component information is retrieved from the database and displayed on the component information page. |

1. **Python Web Scraper API**

|  |  |
| --- | --- |
| Test Level | Integration |
| Quality Criterion | Functionality & Reliability |
| Description of Test | Checking the system to see if Python’s Web Scraper API is scraping the correct data for each component part. |
| Requirements Reference | 1. (Use Case #5) Search for parts with prices, features, etc 2. (Use Case #17) View a PC Information Page |
| Steps of the Test Case | 1. Navigate to the search for parts page 2. Enter a specific part 3. Click on the part 4. Verify if the specific part specifications are correct |
| Expected Outcome | Python’s Web Scraper scraps the correct specifications for each part and adds it into the database. Once a user clicks on a specific part, their correct specifications are displayed. |

1. **Cross Platform Compatibility**

|  |  |
| --- | --- |
| Test Level | Integration |
| Quality Criterion | Portability |
| Description of Test | Ensuring if all the components to QuickPC are functioning in both iOS and Android |
| Requirements Reference | 1. (Use Case #2) Creating an Account 2. (Use Case #3) Login 3. (Use Case #4) View Home Menu 4. (Use Case #7) Cloud Database: User Login Information 5. (Use Case #8) Cloud Database: PC component Information |
| Steps of the Test Case | 1. Open the application on a mobile device 2. Register or log into an existing account 3. Validate if the system is running correctly 4. Verify if functionalities are accessible and usable 5. Repeat steps 1-4 on different mobile devices |
| Expected Outcome | Application and functions are accessible in both iOS and Android |

1. **Price History Graph API**

|  |  |
| --- | --- |
| Test Level | Integration |
| Quality Criterion | Usability |
| Description of Test | Verifies that each component information page includes a correct price history graph. |
| Requirements Reference | 1. (Use Case #5) Search for parts with prices, features, ect 2. (Use Case #8) Cloud Database PC Component Information 3. (Use Case # 17) PC Information Page 4. (Use Case #12) Graph for Price History |
| Steps of the Test Case | 1. Select “Search Part” 2. Select an individual component 3. Verify that the component information page is displaying a valid price history graph. |
| Expected Outcome | User is able to check specific component price history graphs |

# **System Level**

System Level testing ensures that the fully integrated software application is complete and valid. System testing ensures that all conducted tests on a complete application meet the specified requirements for the application system. System testing is done across the entire completed application and makes sure that the application meets the proper standards in regards to performance, workload, efficiency, stability, and reliability of the system application.

1. **PC Component Search Efficiency & Accuracy**

|  |  |
| --- | --- |
| Test Level | System |
| Quality Criterion | Functionality, Efficiency |
| Description of Test | Tests that search queries for particular components retrieve relevant information to the query being made and not retrieve incorrect data |
| Requirements Reference | 1. (Use Case #5) Part Search |
| Steps of the Test Case | Search Bar:   1. Create query for specific PC component based on user input from the search bar 2. Ensure that returned results are relevant to what was being searched based on user input (brands, part names, etc.) 3. Fix searching errors found by QuickPC if any bugs occur 4. Test that search speed is quick and efficient   General Part Search:   1. Create a query for PC components under a general category (RAM, CPU, etc) 2. Ensure that returned results are relevant to what was being searched based on component category 3. Fix searching errors found by QuickPC if any bugs occur 4. Test that search speed is quick and efficient |
| Expected Outcome | Search results return relevant information in context to what is being searched for in a quick fashion |

1. **Efficiency and Clarity of QuickPC (Ease of Use)**

|  |  |
| --- | --- |
| Test Level | System |
| Quality Criterion | Usability, Efficiency |
| Description of Test | Tests that users are able to use QuickPC by themselves: UI and functionality should be relatively intuitive and easy to understand without any external help from outside parties |
| Requirements Reference | 1. (Use Case #3) Login 2. (Use Case #4) View Home Menu 3. (Use Case #5) Part Searching 4. (Use Case #17) PC Information Page 5. (Use Case #13) Component Reviews 6. (Use Case #14) Component Comparison |
| Steps of the Test Case | 1. Test that Sign-up and Log-in pages are easy to use 2. Access all major features from main home page    1. Search for parts    2. Part lists    3. Building guide    4. Current PC Build 3. Log out from application |
| Expected Outcome | QuickPC should be easily accessible to all users at any given moment in time. Users are able to successfully utilize all major features with no trouble. |

1. **Database and Server Requests**

|  |  |
| --- | --- |
| Test Level | System |
| Quality Criterion | Data Security, Reliability |
| Description of Test | Various requests are made to the FireBase, including authentication requests, data retrieval/storage, and user profile retrieval. Simultaneous requests are made to various aspects of the software to ensure usability and functionality across all features. |
| Requirements Reference | 1. (Use Case #2) Sign-up 2. (Use Case #3) Log In 3. (Use Case #5) Part Searching 4. (Use Case #17) PC Information Page 5. (Use Case #13) Component Reviews 6. (Use Case #14) Component Comparison |
| Steps of the Test Case | 1. Create multiple accounts for QuickPC simultaneously 2. Test multiple users logging into their QuickPC accounts at the same time 3. Run multiple requests for part searching simultaneously 4. Run multiple requests for individual part information 5. Write multiple reviews for particular components at a given time 6. Create requests for PC component comparisons |
| Expected Outcome | Server and database are able to satisfy all requests from different users simultaneously without crashing or taking unreasonably long periods of time to execute. |

1. **Data Integrity and Security**

|  |  |
| --- | --- |
| Test Level | System |
| Quality Criterion | Data Security, Integrity, Confidentiality |
| Description of Test | Tests aspects of the system where sensitive user information (emails, passwords) are not exposed to third parties (non-users). Also tests that all data stored on database is not modified by modules within application that shouldn’t modify data |
| Requirements Reference | 1. (Use Case #2) Sign-up 2. (Use Case #3) Log In 3. (Use Case #7) Cloud Database Log-in Information 4. (Use Case #8) Cloud Database PC Component Information 5. (Use Case #17) PC Information Page 6. (Use Case #13) Component Reviews |
| Steps of the Test Case | 1. Log into QuickPC application with authenticated user information 2. Ensure that log in data does not present itself to the user upon login to prevent leaks 3. Search for component parts 4. Click on a component tab to view component information |
| Expected Outcome | All data stored in Firebase is successful. All data for parts is successfully retrieved without unusual modification. |

1. **Server Down Time**

|  |  |
| --- | --- |
| Test Level | System |
| Quality Criterion | Maintainability, Reliability |
| Description of Test | QuickPC servers and databases need to be updated or turned off momentarily for bug fixing/new feature additions |
| Requirements Reference | 1. (Use Case #2) Sign-up 2. (Use Case #3) Log In 3. (Use Case #5) Part Searching 4. (Use Case #17) PC Information Page 5. (Use Case #13) Component Reviews 6. (Use Case #14) Component Comparison |
| Steps of the Test Case | 1. Turn off real-time database and servers 2. Attempt to log into QuickPC to ensure that a “Server Down” message is displayed to users 3. Add necessary fixes/improvements to server/database 4. Ensure that each fix/improvement is working as intended 5. Reboot database |
| Expected Outcome | Anytime servers are down, users should be notified of the server status, and once the server is running again, users should be able to access database/server related features as if nothing happened. |

# 

# **Acceptance Level**

For our acceptance level we had to test QuickPC’s software according to our business requirements to show it meets and fulfills user expectations. For this level user experience is tested for registering an account, logging into your account with correct credentials, searching for PC components, viewing a specific PC part page, and creating a PC build. All of these tests reference use cases from our previous requirements documents and describe the steps of each test case.

1. **Register**

|  |  |
| --- | --- |
| Test Level | Acceptance |
| Quality Criterion | Security, Functional Stability |
| Description of Test | Users will register a new and unique account with an email and password that is authorized by a connected Firebase platform. |
| Requirements Reference | 1. (Use Case #2) Creating an Account 2. (Use Case #7) Cloud Database: User Login Information 3. (Use Case #4) View Home Menu |
| Steps of the Test Case | 1. Launch QuickPC application 2. Tap SignUp button 3. Enter registration information    1. Unique email    2. Password 6 characters or more 4. Tap register button |
| Expected Outcome | Users are directed to the home page of QuickPC where they can access user features with newly created accounts. |

1. **Login**

|  |  |
| --- | --- |
| Test Level | Acceptance |
| Quality Criterion | Reliability, Usability, Functional Stability |
| Description of Test | Returning users with accounts already registered to the cloud database enter login credentials to access the features of QuickPC. |
| Requirements Reference | 1. (Use Case #3) Login 2. (Use Case #7) Cloud Database: User Login Information 3. (Use Case #4) View Home Menu |
| Steps of the Test Case | 1. Launch QuickPC application 2. Enter Login Credentials    1. Email registered on Firebase    2. Password registered to email identifier 3. Tap Login button |
| Expected Outcome | Users are directed to the home page of QuickPC where they can access user features with newly created accounts. |

1. **Search for Parts**

|  |  |
| --- | --- |
| Test Level | Acceptance |
| Quality Criterion | Maintainability, Reliability, Performance Efficiency |
| Description of Test | Users will enter a specific type of part they are looking for and a generated list of parts will be shown to the user. |
| Requirements Reference | 1. (Use Case #5) Search for parts with prices, features, etc 2. (Use Case #7) Cloud Database: PC Component Information 3. (Use Case #17) View a PC Information Page |
| Steps of the Test Case | 1. Navigate to the homepage 2. Select “Search Parts” 3. Select the type of component you are looking for |
| Expected Outcome | Users will be able to look up specific parts they want to view and select a part to view more information about that component. |

1. **View PC Info Page**

|  |  |
| --- | --- |
| Test Level | Acceptance |
| Quality Criterion | Maintainability, Reliability, Usability |
| Description of Test | From a list of generated PC components users will be able to select a PC component, view it’s description, price, and write reviews. |
| Requirements Reference | 1. (Use Case #17) View a PC Information Page 2. (Use Case #7) Cloud Database: PC Component Information 3. (Use Case #13) View Component Reviews |
| Steps of the Test Case | 1. Navigate to the homepage 2. Select “Search Parts” 3. Select a component you are looking for 4. Select a specific part |
| Expected Outcome | Users are able to see the specifications of the specified computer part and leave a review for the specific part. |

1. **Create PC Build**

|  |  |
| --- | --- |
| Test Level | Acceptance |
| Quality Criterion | Maintainability, Functional Stability, Usability |
| Description of Test | Provides users with resources to guide them in assembling a PC and displays a list of the parts that a pc can have, along with the total budget of the parts. |
| Requirements Reference | 1. (Use Case #5) Search for parts with prices, features, etc 2. (Use Case #7) Cloud Database: PC Component Information 3. (Use Case #17) View a PC Information Page |
| Steps of the Test Case | 1. User selects Create PC Build from home menu 2. The user selects a specific PC component 3. List of specific part type that are compatible are displayed for user 4. Specific part can be added to the list and total budget is updated with price of new component |
| Expected Outcome | Builds a list of PC components that the user can view, the total budget of all the parts in the list, and access the PC info page for each part. |