Table of contents

## Concept Paper-------------------------------------------------------------------------------------------------2

## Project Charter------------------------------------------------------------------------------------------------4

## Requirement Gathering And Analysis Document------------------------------------------------------6

## Test Plan------------------------------------------------------------------------------------------------------10

## Test Cases----------------------------------------------------------------------------------------------------14

## Design---------------------------------------------------------------------------------------------------------16

**CONCEPT PAPER**

Project Requester:

Hahu IT School

Statement of the Problem:

Hahu IT School is a private IT school located in Seattle, WA. The school delivers IT courses for students on premises. The school’s objective is to assist in educating IT fields for those who want to enroll and to make them competitive in the industry.

Currently the school tracks it’s student’s and staff’s documents & records on file-based system. Because the system is not automated, getting current information about student’s and staff’s is not easily accessible. The school spends more budget on staff employment for registering students.

Students are unable to access their data online from the school because it currently doesn’t provide an online system.

Project Deliverables and Beneficiaries:

At the end of the project, we hope to have a new system which performs registration of students, login, and finance. In addition, all of the existing data will be migrated to the new system. Access to the system should still be controlled through the use of user IDs and passwords, which are administered by the school.

Strategic Context:

Because the course can be delivered virtually the school can provide students who live on any states.

Project Assumptions and Constraints:

Assumptions:

* The new online system will be accessible from anywhere at any time
* The new online system will make it easier to manage students & staffs

Constraints:

* Subject Matter Expert
* Time

Project Expenses:

**Implementation Expenses:**

There are 2 ways to provide this online service. The first is on the school premises and the second is on the cloud. We recommended to use the cloud because it’s cheaper & there will be no administration of servers. We request the school to provide payment for the cloud provider company $150,000 for 2 years’ service.

We request a total payment of $650,000 for the development of this online system.

**Post-implementation Expenses:**

We include one year of maintenance and support in the purchase price. Beginning in year two maintenance and support are available from the vendor at $20,000.00 annually.

Primary Contact:

Tegenne Gebere, Hahu school owner: 295-7291 TegenneG@gmail.com

**Project charter**

1. **General Information**

|  |  |  |  |
| --- | --- | --- | --- |
| ***Project Title:*** | Hahu IT School Digitalization Project | | |
| ***Brief Project Description:*** | To establish a web-based platform that helps a school take student’s data online for an easier management | | |
| ***Prepared By:*** | Elite Technologies | | |
| ***Date:*** | 06/14/2022 | ***Version:*** | 1.0 |

1. **Project Objective:**

* At the end of the project, we hope to have a new system which performs registration of students, record management, finance.

1. **Assumptions**

* The new online system will be accessible from anywhere at any time
* The new online system will make it easier to manage students & staffs

1. **Project Scope**

* Registration of Students & Staffs
* Record Management
* Finance

1. **Roles and Responsibilities**

|  |  |
| --- | --- |
| **Sponsor:** | |
| **Name** | **Email / Phone** |
| * Tegenne Gebre | [TegenneG@gmail.com](mailto:TegenneG@gmail.com) |
| **Project Manager:** | |
| **Name** | **Email / Phone** |
| Binyam Birhane |  |
| **Team Member:** | |
| **Name** | **Email / Phone** |
| Misgana Abreham System Analyst |  |
| Yohana Dessalegn System Analyst |  |
| Rose Kussa Developer |  |
| Nebeyu Yilma Developer |  |
| Mikiyas Berta Tester |  |
| Eyerusalem Shiferaw Tester |  |
| **Customer:** | |
| **Name** | **Email / Phone** |
| Hahu IT School |  |

**F. Resources**

|  |  |
| --- | --- |
| **Resource** | **Constraints** |
| Hahu IT School | $820,000 |

**G. Project Risks**

|  |  |
| --- | --- |
| **Risk** | **Mitigation Strategy** |
| Time | Add additional technical persons |
| Subject Matter Expert | Observation, Interview & brain storming |

**H. Signatures**

|  |  |  |
| --- | --- | --- |
| **Customer:** |  |  |
| **Name** | **Signature** | **Date** |
|  |  |  |
|  |  |  |
| **Project Sponsors:** |  |  |
| **Name** | **Signature** | **Date** |
|  |  |  |
|  |  |  |
| **Project Manager:** |  |  |
| **Name** | **Signature** | **Date** |
|  |  |  |
|  |  |  |

**Requirement Gathering and Analysis Document**

1. **Introduction**
   1. **Purpose of Document**

This is a requirement gathering and analysis document for Hahu It school. This school isn’t automated. This document describes the scope, objectives and goal of the new system. In addition to describing non-functional requirements, this document models the functional requirements with use cases, interaction diagrams, and class models.

* 1. **Project Summary**

Project Name: Hahu It School Digitalization

Project Manager: Binyam

Project Analysts: yohana, misgana

Responsible users: Ato Tegenne

1. **Functional Objectives**
2. The system shall allow students for online registration.
3. The system shall display the available courses and updated information.
4. The system shall allow to make an online payment.
5. The system allows the school’s staff to directly contact with the students via email or phone number.
6. **Non-Functional Objectives**

**3.1 Reliability**

* The system shall be completely operational at least 90% of the time.
* Down time after a failure shall not exceed 2 hours.

**3.2 Usability**

* A user can access the system easily.

**3.3 Performance**

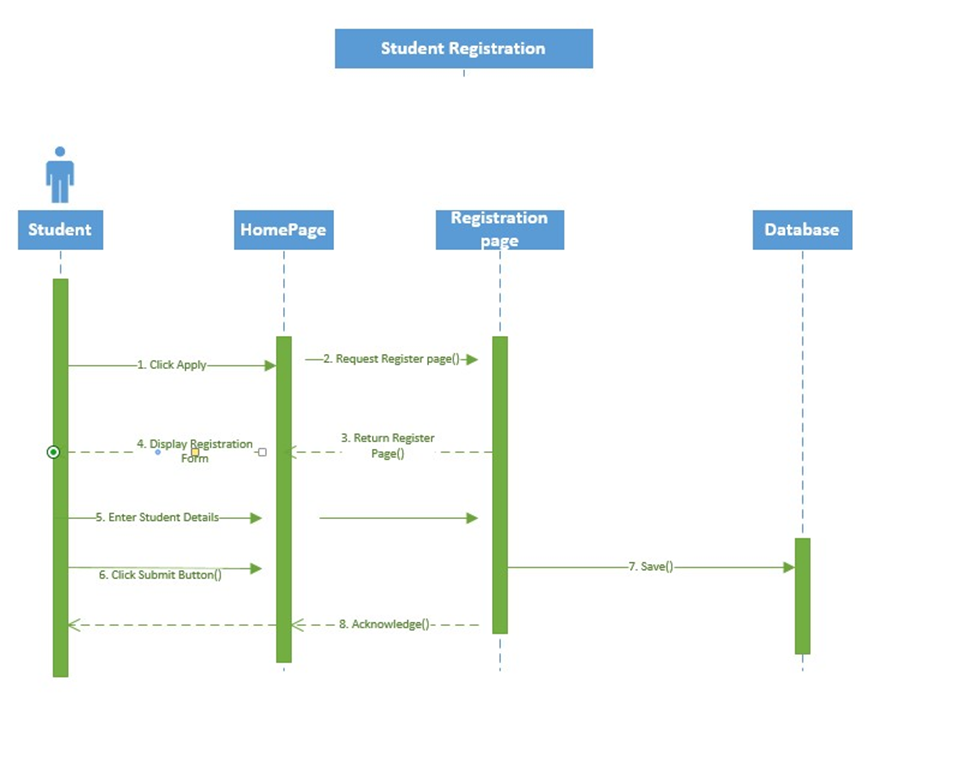
* The system should be able to support 200 simultaneous users.

**3.4 Supportability**

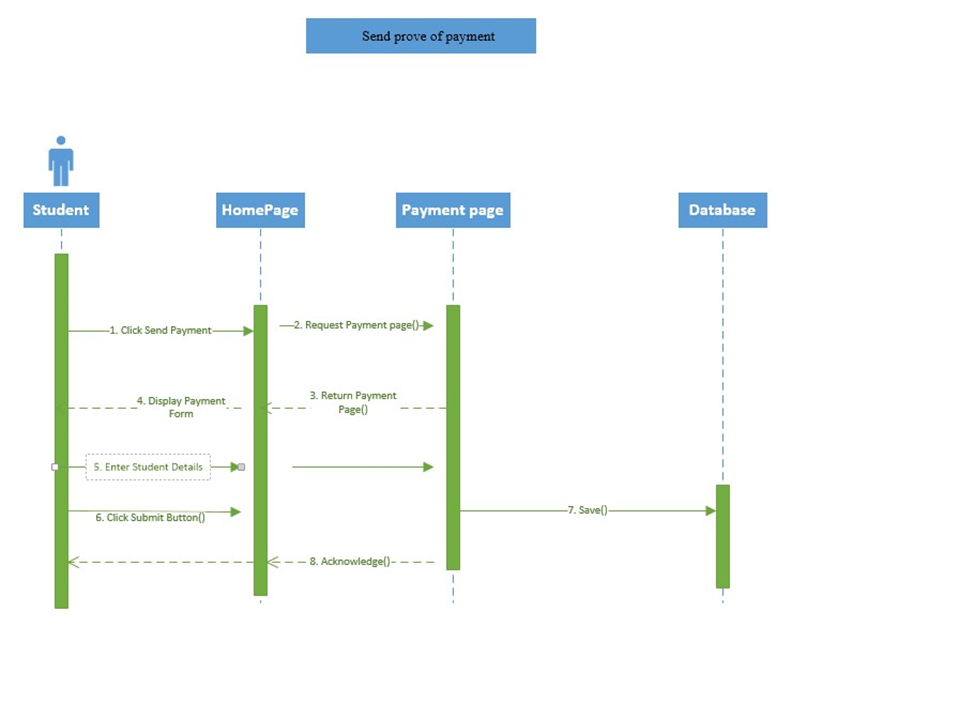
* The system should be able to accommodate new courses and references without major reengineering.

1. **Use Case Model**

|  |  |
| --- | --- |
| Use Case Name: | Register User |
| Summary: | To get accepted for a course, the student should first register |
| Basic flow: | 1. The use case starts when the student wants to apply 2. The system requests first name, last name, middle name, email, phone number, sex (male or female), country (USA), state, city, highest education level, highest education category, higher education field of study, program of interest, select your preferred Schedule. 3. The system sends either an approval or denial statement via email. |
|  |  |
| Precondition: | None |
| Postcondition: | Student data will be saved on the database |



|  |  |
| --- | --- |
| Use Case Name: | Send prove of payment |
| Summary: | To get attained in a class, a student should send prove of payment |
| Basic flow | * + 1. The case started when the student transfers the fee & want to send the prove     2. The system request First name, last name, class name, amount     3. Attached document     4. Acknowledge |
|  |  |
| Precondition: | The student should be accepted by the school |
| Postcondition: | Student data will be saved on the database |



**Test Plan**

1. **INTRODUCTION (Executive Summary)**

The Elite group preparing Software Test plan (STP) for HaHU IT online School**,** to be granted, verify and deliver the completed software. By using varies methods of tests and will projected the results. And discussing with the expertise in the team, scheduling a time for each phase the test will be completed.

**2.0 OBJECTIVES AND TASKS**

**2.1 Objectives**

This document describes the plan for testing the overall design of the HaHu online school System. And in the long term to ensure the quality of the software and the user expectations are satisfied.

**2.2 Tasks**

This Test Plan document supports the following objectives:

* + Identify project information/or requirements and the software that should be tested.
  + List the recommended test requirements.
  + Recommend and describe the testing strategies/Methods to be applied.
  + Identify the required resources.
  + List the deliverable elements of the test activities.

**3.0 SCOPE**

This Test Plan will use and describes the **integration** and **system** tests that will be conducted on the project following integration of the subsystems and components identified in the Integration Build Plan for the System.

It is assumed that **unit testing** will be held with developers in extensive coverage of source code, and testing of all module interfaces.

The purpose of reviewing the requirements is to test feasibility and performance of the software system. It is critical that all projects should be tested as well as system performance at this early stage. And Testing of system functionality and features will be conducted on the project.

The interfaces between the following subsystems will be tested:

* 1. Application
  2. Course Registration
  3. Payment System
  4. Course List/Catalog.

The external interfaces to the following devices will not be tested:

* 1. Local PCs
  2. Remote PCs.

The most critical performance measures to test are:

* 1. Response time for remote login to the Application and course registration system.
  2. Response time to access the Payment System and to upload the image.
  3. Student response time when system loaded with 100 logged in students.
  4. Student response time when 15 simultaneous accesses to the Course Catalog database.

**4.0****Requirements for Test**

The listing below identifies those items (use cases, functional requirements, non-functional requirements) that have been identified as targets for testing. This list represents *what* will be tested.

**4.1 Data and Database Integrity Testing**

* Verify access to Programs Database.
* Verify simultaneous record read accesses.
* Verify correct retrieval of update of database data.

**4.2 User Interface Testing**

* Verify ease of navigation through a sample set of screens.
* The desktop user-interface shall be Windows OS compliant.
* The user interface of the Registration System shall be designed for ease-of-use and shall be appropriate for IT user community with no additional training on the System.

**4.3 Performance Testing**

* Verify response time for login.
* Verify response time for submittal of registration.
* Verify response time to access programs system and payment system.

**4.4 Load Testing**

* Verify system response when loaded with 100 logged on students.
* Verify system response when 15 simultaneous student accesses to the Programs.

**4.5 Security and Access Control Testing**

* Verify Logon from a local PC.
* Verify Logon security through username and password mechanisms.

**4.6 Configuration Testing**

* The system shall run on Windows OS.
* The web-based interface for the System shall run in Google Chrome, Mozilla and Internet Explorer browsers.

**5.0****Test Strategy**

The Test Strategy presents the recommended approach to the testing of the software applications. It gives answer to How it will be tested.

|  |  |  |  |
| --- | --- | --- | --- |
| **Testing Type** | **Test Objectives** | **Completion Criteria** | **Special Consideration** |
| Data and Database Integrity Testing | Ensure Database access methods and processes function properly and without data corruption by activating each database access method and process, entering each with valid and invalid data. | All database access methods and processes function as designed and without any data corruption |  |
| Function Testing | |  | | --- | | Ensure proper application navigation, data entry, processing, and retrieval. | | |  | | --- | | All planned tests have been executed and all identified defects have been addressed. | |  |
| Security and Access Control Testing | Function / Data Security: Verify that user can access only limited functions / data for which their user type is provided permissions.  System Security: Verify that only those users with access to the system and application(s) are permitted to access them. | For each known user type the appropriate function / data are available and all transactions function as expected and run-in prior Application Function tests |  |
| Failover and Recovery Testing | The entirety of our unit testing will be considered complete once all test cases pass successfully, and we have achieved at least 70% code coverage. | If it is impossible to continue testing due to some defects, we will suspend testing until the defects are eliminated. |  |
| Configuration Testing | Validate and verify that the client Software function properly. | successfully completed without failure. |  |

1. **Responsibilities**

|  |  |  |
| --- | --- | --- |
| **Name** | **Position** | **Responsibility** |
| Mikias Berta | Lead Tester | - Lead, guide and monitor the analysis, design, implementation and execution of the test cases, test procedures and test suites.  - During test execution and as the project winds down, they write summary reports on test status. |
| Eyerusalem Shiferaw | Tester | -Reviewing software requirements and preparing test scenarios.  - Executing tests on software usability.  -Interacting with clients to understand product requirements |

### **7.0 Schedule**

|  |  |  |  |
| --- | --- | --- | --- |
| **Milestone Task and Deliverable** | **Start Date** | **End Date** | **Due Date** |
| Test Plan | 6/27/2022 | 6/29/2022 | June 29th, 2022 |
| Test Case Designing and Development | 6/29/2022 |  | July 1st 2022 |
| Test Environment Setup |  |  | July 4th 2022 |
| Test Execution |  |  | July 6th 2022 |
| Test Closure |  |  | July 6th 2022 |

**8.0 References**

Applicable references are:

* Requirement Gathering Documents
* Use-case diagram
* Azure Devops

**Test Cases**

**Registration**

Verify that a user with long First name can apply

Verify that create an account link is present or not

Verify that create an account link is clickable or not

Verify that a user with long Last name can apply

Verify a user without valid email is not able to apply

Verify that all slots are filled except middle name before clicking next

Verify that incase a user forgets to fill in a slot, there's a message display underneath that notifies the user

Verify that there is First name field in the personal information section

Verify that there is Last name field in the personal information section

Verify all the mandatory fields in the registration page are marked with red color \* symbol

Verify that the right font size is displayed

Verify that next and back buttons are present

Verify that all the field have a valid placeholder

Verify whether the fields in the registration page are according to the client's requirement

Verify that all buttons work and accessible to the user

Verify that tab functionality is working properly or not

Verify that the response time after clicking ‘Apply’ is less than 1 second

Verify that multiple users can use the registration page all at once

Verify the response time after filling out the personal information is less than 1 second

Verify that SSN is not asked on the personal information section

Verify there's an expiration time limit on the registration page if left open for more than 10 minutes

Verify there's a login page present or not

**Payment**

Verify that first name is present

Verify that last name is present

Verify that on the amount field, number and symbols is accessible

Verify whether the ‘upload file’ on the Proof of payment field is clickable

Verify all buttons and text boxes are visible and clickable

Verify all mandatory fields are marked with red color and a symbol \*

Verify that if a user leaves text box blank, there’s a message display that says “this question is required”

Verify the submit button is visible to the user

Verify the submit button is clickable

Verify all the options for the program field are clickable

Verify proof of payment can successfully be uploaded

**Courses**

Verify all given courses are available to view to users

Verify that all courses have summarized detail underneath the course name

Verify that all courses are clearly written with the right font size

Verify that 100 users will be able to view the course field at the same time

Verify that there’s a comment section for users to comment or ask any questions

Verify that all fields' buttons are clickable

Verify that there is course schedule present

**Meet the team**

Verify that meet the team field button is clickable

Verify that the drop-down menu is clear and easy to read to users

Verify that in the drop-down menu, there is ‘about us’ field

Verify that in the drop-down menu, there is ‘contact us’ field

Verify that there is upcoming events field is listed

Verify that there is staff/instructor field present

Verify that there is direction is listed

Verify that Instructor field is clickable

Verify all instructors have their name available to see to users

Verify that underneath of all the instructors name, there is a course name in what they attended to

Verify that there is a picture of all the instructors available to see to users

**Design**

Application page design

|  |  |
| --- | --- |
| First Name |  |
| Last Name |  |
| Middle Name |  |
| Email |  |
| Phone Number |  |
| sex |  |
| * female | * male |
| Higher education level   * high school * college student * diploma * bachelor * Masters * Doctorate |  |
| Highest education Category   * Natural science * Social science * Engineering * Legal * Information technology * Business * Health science * Other |  |
| Higher Education Field of study |  |
| Program Of Interest   * Software Test Engineers * Database Developer * Business Intelligence * Beginner IT Professional * System Analyst * Web Developer * Data Warehouse Developer |  |
| Select Your Preferred Schedule   * Monday and Wednesday: 6:00 pm-9:00 pm * Friday and Saturday: 6:00 pm-9:00Pm |  |

Personal Information and educational information table

Address Table

|  |  |
| --- | --- |
| Country | USA |
| **State**  • Alabama AL  • Alaska AK  • Arizona AZ  • Arkansas AR  • California CA  • Colorado CO  • Connecticut CT  • Delaware DE  • Florida FL  • Georgia, GA  • Hawaii HI  • Idaho ID  • Illinois IL  • Indiana IN  • Iowa IA  • Kansas KS  • Kentucky KY  • Louisiana LA  • Maine ME  • Maryland MD  • Massachusetts MA  • Michigan MI  • Minnesota MN  • Mississippi MS  • Missouri MO  • Montana MT  • Nebraska NE  • Nevada NV  • New Hampshire NH  • New Jersey NJ  • New Mexico NM  • New York NY  • North Carolina NC  • North Dakota ND  • Ohio OH  • Oklahoma OK  • Oregon OR  • Pennsylvania PA  • Rhode Island RI  • South Carolina SC  • South Dakota SD  • Tennessee TN  • Texas TX  • Utah UT  • Vermont VT  • Virginia VA  • Washington WA  • West Virginia WV  • Wisconsin WI  • Wyoming WY |  |
| City |  |
| **Submit** | |

**Home Page Design**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Login** | **Programs** | **About Us** | **Registration** | **Resources** | **Submitpayment** | **help** |