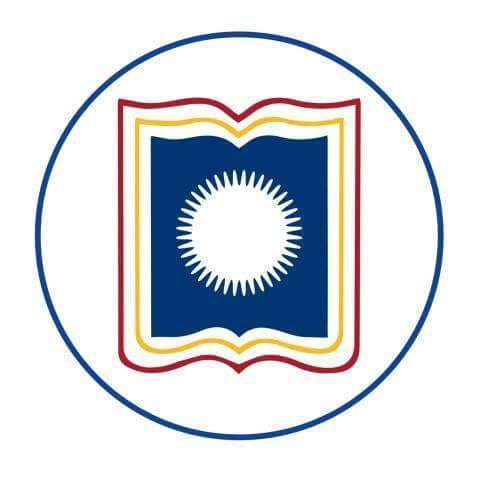
**NilamHut**



**Rajshahi University**

**Computer Science & Engineering**

**Department**

**B.S. Engineering Project-1**

**A proposal for making an auction platform for Bangladesh**

**Session: 2013-14**

**Part-4, Even Semester**

|  |  |
| --- | --- |
| **Name of the Students** | **Rajob Raihan Monmoy, Jaggesher Mondal** |
| **ID Number** | **140754081, 14025423** |
| **Name of the Supervisor** | **Associate professor Md. Morshedul Arifin** |

**Abstract**

In Bangladesh there are no good auction sites. There are no good place for auctioning rare as well as common products. The process of auction is not digitized. The auction arranger calls for an auction in specific place and people who are interested attends there physically. In current system arranging auction is not easy. The process of notifying people about auction is lengthy and difficult process. Sometimes it’s hard for people to join auction physically. So we will automate the process of auctioning in Bangladesh so that people will not have to join the auction physically and the auction arranger can arrange auction easily. We will make use of single page application for fluid and fast user experience. User can join auction digitally with being physically going to the auction site. This is much more convenient. For example Government can arrange tender in the auction platform.

|  |  |  |
| --- | --- | --- |
| **Chapter 1 : Introduction** | | |
| 1.1 | Introduction | 01 |
| 1.2 | Overview | 02 |
| 1.3 | Scope of the Project | 03 |
| **Chapter 2 : Background Study and Planning** | | |
| 2.1 | Background Study | 01 |
| 2.2 | Core Objective | 01 |
| 2.3 | Software Model | 01 |
| 2.4 | Planning |  |
| **Chapter 3 : Feasibility Study** | | |
| 3.1 | Technical Feasibility | 01 |
| 3.2 | Economic Feasibility | 01 |
| 3.3 | Environmental Feasibility | 01 |
| 3.4 | Schedule Feasibility |  |
| 3.5 | Application Area |  |
| **Chapter 4 : Design Implementation** | | |
| 4.1 | Technology to be Used | 01 |
| 4.2 | Use Case(UML) Diagram |  |
| 4.3 | ER Diagram |  |
| **Chapter 5 : Risk Factor And Management** | | |
| 5.1 | Risk Factor | 01 |
| 5.2 | Risk Management | 01 |
| **Chapter 6 : Testing** | | |
|  |  |  |
| **Chapter 7 : Conclusion** | | |
|  |  |  |
| 1.3 | Scope of the Project | 03 |

Chapter [1]

*Introduction*

**1.1 Introduction**

In Bangladesh there are no good auction sites. There are no good place for auctioning rare as well as common products. The process of auction is not digitized. The auction arranger calls for an auction in specific place and people who are interested attends there physically. In current system arranging auction is not easy. The process of notifying people about auction is lengthy and difficult process. Sometimes it’s hard for people to join auction physically. So we will automate the process of auctioning in Bangladesh so that people will not have to join the auction physically and the auction arranger can arrange auction easily. We will make use of single page application for fluid and fast user experience. User can join auction digitally with being physically going to the auction site. This is much more convenient. For example Government can arrange tender in the auction platform.

**1.2 Overview**

NilamHut is a place where products are sold through Auction. Live auction can be done on posted product for given period of time. Users can attend those auctions online. Users can view complete product description, view the user who posted the auction, user can also see the current top bid as well as the person who has done the bid. User can see top 10 bids and the bidding is live broadcasted to all the user who is currently in the website. Proper authentication, authorization and security features will be provided by the site. For example people who arranged the auction can’t bid in that product, current bid must be higher than the base value and previous highest bid, only admin can add the category, city and tags etc. There will be rating for users and poster. User and auction arranger can see the rating and know about the credibility of the person. For security purpose we implemented HTTPS. This platform will be made for Bangladesh.

**1.3 Scope of the Project**

The scope of the project is to build an online platform specifically for auction purpose in Bangladesh. It also serves as an E-Commerce platform. Live auction, secure transaction, fluid and fast user experience, finding products using different search parameters easily, secure web surfing, creating a complete business platform is the final goal of the project. Implementing HTTPS, Socket programming, SPA, JWT, API are needed for the success of the project. We will need to separately implement front end and back end for the fast page rendering of the project.

Chapter [2]

*Background Study and planning*

**2.1 Background Study**

* **Local Market**
* There are no auction sites in Bangladesh.
* Selling products on international auction sites is not practical.
* International auctions platforms like beezid[1], wellbid[2] is not available in Bangladesh.
* **International Market**
* International auctions platforms like beezid[1], ebay[3] does not support real time auction.
* Most of the international platform is unreliable and does not use SPA.

**2.2 Core Objective**

In Bangladesh there are multiple platforms for online selling but there are no platform which implements selling via auction, which is very important for some products. Here is our final goals.

* Auctions can be done on previously posted products.
* Live Auction.
* Fast and fluid user experience.
* User rating system.
* Users will be able to see the number of bids and numbers of auction a person has done.
* Bidding is live broadcasted to all the user who is currently in the website.
* Users will be able to see top 10 bids without refreshing the page.
* Finding products using different search parameters.
* Secure transaction via NilamHut.

**2.3 Software Model**

The waterfall model is a sequential (non-iterative) design process, used in software development processes, in which progress is seen as flowing steadily downwards (like a waterfall) through the phases of conception, initiation, analysis, design, construction, testing, production/implementation and maintenance.

Requirement Definition

Coding

Integration & Test

Modeling

Design

Planning

Designing & Scheduling

Maintenance &

Support

This model some time called the classic life cycle, suggest a systematic, sequential approach to software development which begins with user specification of requirements and process through planning, modeling construction, and deployment , culminating in ongoing support of the complete software. The main approach of this model is much similar to a real waterfall that’s it called waterfall model.

**Why Choose Waterfall Model:**

Now a days we easily notice that waterfall model strong enough for most of the software industries because of its some limitation but for our purpose it is completely suitable. So here is the reasons why we choose this model except other:

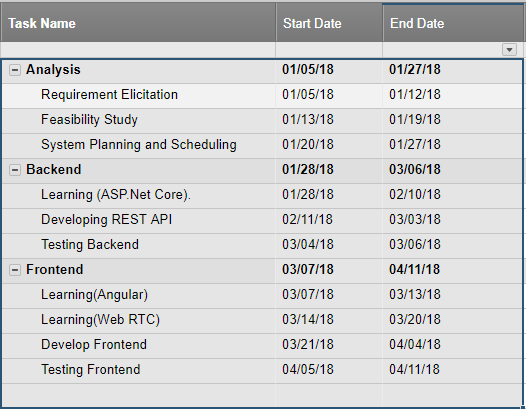
1. We wanted simple and easy to use model. Waterfall model is simple and easy to use.
2. Requirement for the system are very well understood, clear and fixed.
3. Our product definition is stable.
4. There are no ambiguous requirements in the system.
5. The project is not that big.
6. Easy to manage due to rigidity of the model.
7. When an error is identified we can instantly loop back to previous stage and problem is solved. So this is a more secure model for us.

**2.4 Planning**

The planning indicates what needs to be done, which resources must be utilized, and when the project is due. In short, it’s a timetable that outlines start and end dates and milestones that must be met for the project to be completed on time. Project planning envelops the following actions:

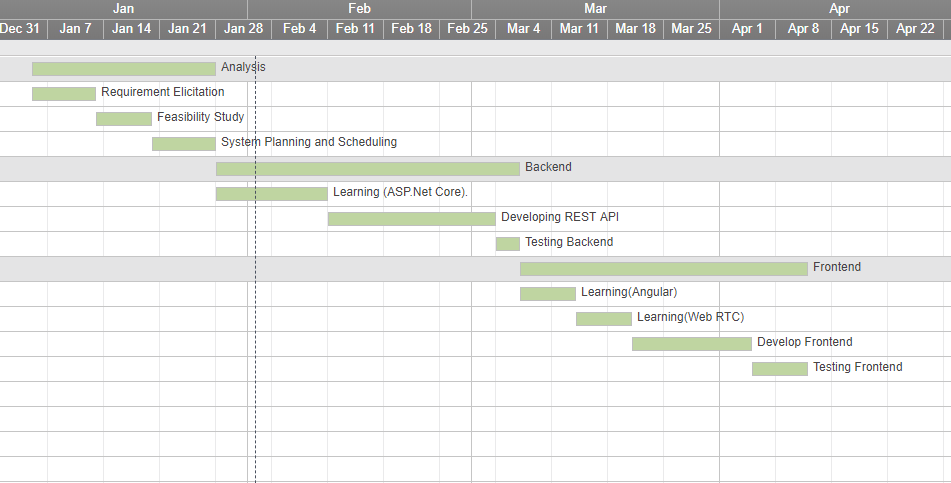
1. Split project into task and estimate time and resources required to complete each task.
2. To make the optimal use of workforce, we have to organize tasks concurrent.
3. It depends on project manager’s skill and experience.

The **task list** and require time for this project is given below:



A Gantt chart, commonly used in project management, is one of the most popular and useful ways of showing activities (tasks or events) displayed against time. On the left of the chart is a list of the activities and along the top is a suitable time scale. Each activity is represented by a bar; the position and length of the bar reflects the start date, duration and end date of the activity.

The **gantt chart** for developing the project is given below:



Chapter [3]

*Feasibility Study*

**3.1 Technical Feasibility**

Technical feasibility, studies that if the current technological infrastructure are capable of accommodating candidate system. The [technical feasibility](https://en.wikipedia.org/wiki/Technical_feasibility) assessment is focused on gaining an understanding of the present technical resources of the organization and their applicability to the expected needs of the proposed system. It is an evaluation of the hardware and software and how it meets the need of the proposed system.

* We have experience in ASP.Net Core on backend and Angular on front end.
* We have also experience on web RTC for real time communication.
* User can use the browser from mobile, tablet or pc to browse our web site. All modern browsers are capable of running this site.

**3.2 Economic Feasibility**

The purpose of the economic feasibility assessment is to determine the positive economic benefits to the organization that the proposed system will provide. It includes quantification and identification of all the benefits expected. This assessment typically involves a cost/benefits analysis.

* There are no competitor site. So benefits are expected to come shortly.
* NilamHut will charge 5% for each sale. So it is profitable.
* User can attend or bid on products from anywhere. User don’t have to be there physically.

**3.3 Environmental Feasibility**

It is important to understand and analysis how the users (General People) will react towards the new online digital system. And here there is a question that is this design are capable to satisfy all users. Of course there here is a majority issue but the fundamental thing is to concern that the objective of the system to be design.

* NilamHut will provide secure transaction.
* NilamHut will prevent frauds.
  1. **Schedule Feasibility**

Schedule feasibility is define as the likelihood of a project begin completed within its time frame. It is necessary to determine whether the deadlines are mandatory or desirable.

* We will be able to complete the project in the time schedule.

**3.5 Application Area**

* Government and other organizations like charity organization can use NilamHut for auction.
* General people can also sell their valuable and rare products.
* People can bid online regardless of the place.

Chapter [4]

*Implementation*

**4.1 Technology to be used**

* **Server Side**
  + Language
    - C#
  + Framework
    - ASP.Net Core
  + Database
    - Sql Server
* **Client Side**
  + HTML
  + CSS
  + Bootstrap 4
  + Angular (Angular 6)
  + Angular UI
* **ASP.NET CORE SignalR for real time broadcasting.**

**HTML:**

HTML is a computer language devised to allow website creation. These websites can then be viewed by anyone else connected to the Internet. It is relatively **easy to learn**, with the basics being accessible to most people in one sitting; and quite **powerful** in what it allows you to create. It is constantly undergoing revision and evolution to meet the demands and requirements of the growing Internet audience under the direction of the W3C, the organization charged with designing and maintaining the language.

**CSS:**

**C**ascading **S**tyle **S**heets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable.

CSS handles the look and feel part of a web page. Using CSS, you can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, layout designs, variations in display for different devices and screen sizes as well as a variety of other effects.

CSS is easy to learn and understand but it provides powerful control over the presentation of an HTML document. Most commonly, CSS is combined with the markup languages HTML or XHTML.

**Bootstrap:**

Bootstrap is an open-source JavaScript framework developed by the team at Twitter. It is a combination of HTML, CSS, and JavaScript code designed to help build user interface components. Bootstrap was also programmed to support both HTML5 and CSS3.

Also it is called Front-end-framework.

Bootstrap is a free collection of tools for creating a websites and web applications.

It contains HTML and CSS-based design templates for typography, forms, buttons, navigation and other interface components, as well as optional JavaScript extensions.

Some Reasons for programmers preferred Bootstrap Framework

* Easy to get started
* Great grid system
* Base styling for most HTML elements(Typography,Code,Tables,Forms,Buttons,Images,Icons)
* Extensive list of components

**ASP.NET CORE**:

ASP.NET Core is a cross-platform, high-performance, open source framework for building modern, cloud-based, Internet-connected applications. With ASP.NET Core, you can:

* Build web apps and services, IOT apps, and mobile backends.
* Use your favorite development tools on Windows, macOS, and Linux.
* Deploy to the cloud or on-premises.
* Run on .NET CORE and .NET Framework.

**Why use ASP.NET CORE**:

Millions of developers have used (and continue to use) this to create web apps. ASP.NET Core is a redesign of ASP.NET 4.x, with architectural changes that result in a leaner, more modular framework.

ASP.NET Core provides the following benefits:

* A unified story for building web UI and web APIs.
* Integration of modern, client side frameworks  and development workflows.
* A cloud-ready, environment-based configuration systeam.
* Built-in dependency injection.
* A lightweight, high performance, and modular HTTP request pipeline.
* Ability to host on IIS, NGINX, Apache, Docker or self-host in your own process.
* Side-by-side app versioning when targeting .Net Core .
* Tooling that simplifies modern web development.
* Ability to build and run on Windows, macOS, and Linux.
* Open-source and community focused.

**Angular**:

Angular is a JavaScript framework that helps developers build applications. The library provides a number of features that make it trivial to implement the complex requirements of modern applications, such as data binding, routing, and animations.

Angular also provides a series of conventions for how you approach application development, which can be beneficial for large teams that need to work together on a single code base. Angular is one of the only JavaScript libraries to provide a comprehensive style guide with a number of opinionated guidelines on how you could write your code with the framework.

**Why use Angular:**

Single-Page Applications (SPAs) are Web apps that load a single HTML page and dynamically update that page as the user interacts with the app.

SPAs use AJAX and HTML5 to create fluid and responsive Web apps, without constant page reloads. However, this means much of the work happens on the client side, in JavaScript. For the traditional ASP.NET developer, it can be difficult to make the leap. Luckily, there are many open source JavaScript frameworks that make it easier to create SPAs. We choose Angular because:

* Out of box support for navigation (with almost every other framework, you need a separate library for this).
* Support for data maintenance using services.
* Support for lazy loading.
* Loads of inbuilt filters, pipes for modulating your data on UI.
* Comes with complete server configuration for development mode
* Many Options like Guards, resolvers, directives and pipes make your life hell lot easier.
* Styles encapsulation and component scopes makes it lot easier to maintain your stylings.
* Angular6 is due to release soon and promises support for progressive web apps.

**SignalR:**

ASP.NET SignalR is a new library for ASP.NET developers that makes developing real-time web functionality easy. SignalR allows bi-directional communication between server and client. Servers can now push content to connected clients instantly as it becomes available. SignalR supports Web Sockets, and falls back to other compatible techniques for older browsers. SignalR includes APIs for connection management (for instance, connect and disconnect events), grouping connections, and authorization.

**Why use** **SignalR:**

With SignalR, the server can call a JavaScript methods on all the clients by itself when updates are required. The library will handle the connection needed to achieve this: by default WebSocket is used, but it will fallback automatically to older connections types if WebSocket is not available in the browser. The JavaScript can also call the server: this can already be done with AJAX, but if two-way communication is needed it may be easier and cleaner to do it all with SignalR.

So, using a real-time library is the way to go if you want to build an application that requires collaboration between users. Common uses cases includes editors, social networks, chats or a bid broadcasting like our project.

**4.2 UML (Use Case) Diagram**