Feasibility Evidence Description (FED)

Meta Nutrition Blog

Team 05

Adit Dharia  Project Manager
Aditya Shinde  System and Software Architect
Akshay Jain  Feasibility Analyst
Aman Mahajan  Prototyper
Arnav Jalui  Requirements Engineer
Ayush Jain  Implementer
Elenaz Ramezani  Operational Concept Engineer
Soham Chitalia  Prototyper

October 20th, 2020
## Version History

<table>
<thead>
<tr>
<th>Date</th>
<th>Author</th>
<th>Version</th>
<th>Changes made</th>
<th>Rationale</th>
</tr>
</thead>
<tbody>
<tr>
<td>10/01/20</td>
<td>SC</td>
<td>1.0</td>
<td>· Original template</td>
<td>· Started with original template from class website</td>
</tr>
<tr>
<td>10/02/20</td>
<td>SC, AJ</td>
<td>1.1</td>
<td>· Add section 1</td>
<td>· Created initial changes and delegated work to relevant team members</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>· Add stakeholder info</td>
<td></td>
</tr>
<tr>
<td>10/02/20</td>
<td>AJ</td>
<td>1.1.1</td>
<td>· Add Cost Analysis</td>
<td></td>
</tr>
<tr>
<td>10/02/20</td>
<td>SC</td>
<td>1.1.2</td>
<td>· Add benefit analysis</td>
<td>· Adding benefit analysis after team meeting with client</td>
</tr>
<tr>
<td>10/03/20</td>
<td>SC, AJ</td>
<td>1.2</td>
<td>· Add ROI calculations and graph</td>
<td>· Add ROI based on Cost and Benefit analysis</td>
</tr>
<tr>
<td>10/04/20</td>
<td>AJ</td>
<td>1.3</td>
<td></td>
<td>· Add Architecture Feasibility</td>
</tr>
<tr>
<td>10/05/20</td>
<td>SC</td>
<td>1.4</td>
<td>· Add process feasibility</td>
<td></td>
</tr>
<tr>
<td>10/06/20</td>
<td>AJ, SC</td>
<td>1.5</td>
<td>· Add risk assessment</td>
<td>· Risk assessment section added based on biweekly Risks and Defect worksheet</td>
</tr>
<tr>
<td>10/07/20</td>
<td>AJ</td>
<td>1.6</td>
<td>· Add NDI/NCS interoperability assessment</td>
<td></td>
</tr>
<tr>
<td>10/08/20</td>
<td>AJ, SC</td>
<td>2.0</td>
<td>· Changes based on team review</td>
<td>· Review and create initial draft for team meeting and discussion</td>
</tr>
<tr>
<td>Date</td>
<td>Name(s)</td>
<td>Version</td>
<td>Changes</td>
<td></td>
</tr>
<tr>
<td>------------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td></td>
</tr>
<tr>
<td>10/10/20</td>
<td>AJ, SC</td>
<td>2.5</td>
<td>Changes based on review meeting with team</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>· Add suggestions and changes from other team members in preparation for ARB</td>
<td></td>
</tr>
<tr>
<td>10/12/20</td>
<td>SC</td>
<td>2.6</td>
<td>Changes to capability feasibility section 3.2</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>· New requirements introduced by client to be included in capability feasibility section</td>
<td></td>
</tr>
<tr>
<td>10/14/20</td>
<td>AJ</td>
<td>2.7</td>
<td>Changes to risk assessment section 5</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>· Change in client requirements and incomplete UI designs provided by client</td>
<td></td>
</tr>
<tr>
<td>10/14/20</td>
<td>SC</td>
<td>2.8</td>
<td>Changes to evolutionary feasibility section 3.3</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>· Changes made due to change in client requirements</td>
<td></td>
</tr>
<tr>
<td>10/18/20</td>
<td>AJ</td>
<td>2.9</td>
<td>Changes to section 3.1 LOS table after client discussion</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>· Changing the LOS requirements for availability of the website</td>
<td></td>
</tr>
<tr>
<td>10/19/20</td>
<td>AJ, SC</td>
<td>3.0</td>
<td>Changes to section 6.2 based on ARB presentation</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>· Adding detailed analysis of COTS product used</td>
<td></td>
</tr>
</tbody>
</table>
Table of Contents

Feasibility Evidence Description (FED) ........................................................................................................ 1
Version History ............................................................................................................................................. 2
Table of Contents ........................................................................................................................................ 4
Table of Figures ............................................................................................................................................ 6
1. Introduction .............................................................................................................................................. 7
   1.1. Purpose of the FED Document ........................................................................................................... 7
   1.2. Status of the FED Document .............................................................................................................. 7
2. Business Case Analysis ............................................................................................................................ 8
   2.1. Cost Analysis ....................................................................................................................................... 9
      2.1.1. Personnel Costs ............................................................................................................................ 9
      2.1.2. Costs associated with deployment on cloud services ................................................................. 10
   2.2. Benefit Analysis ................................................................................................................................ 10
   2.3. ROI Analysis ..................................................................................................................................... 12
3. Architecture Feasibility ............................................................................................................................ 15
   3.1. Level of Service Feasibility ............................................................................................................... 15
   3.2. Capability Feasibility .......................................................................................................................... 17
   3.3. Evolutionary Feasibility .................................................................................................................... 18
5. Risk Assessment ....................................................................................................................................... 21
6. NDI/NCS Interoperability Analysis ......................................................................................................... 24
   6.1. Introduction ....................................................................................................................................... 24
      6.1.1. COTS / GOTS / ROTS / Open Source / NCS ........................................................................... 24
      6.1.2. Connectors ................................................................................................................................... 24
      6.1.3. Legacy System ............................................................................................................................. 25
   6.2. Evaluation Summary .......................................................................................................................... 25
Table of Tables

Table 1: Business Case Analysis ........................................................................................................... 8
Table 2: Personnel Costs .......................................................................................................................... 9
Table 3: Deployment Costs - Operation .................................................................................................. 10
Table 4: Benefits of New Meta Nutrition Blog System ........................................................................ 11
Table 5: Visitor Trend of Meta Nutrition Blog System ........................................................................ 11
Table 6: ROI Analysis ............................................................................................................................... 12
Table 7: ROI Analysis with software cost ............................................................................................ 14
Table 8: Level of Service Feasibility ...................................................................................................... 15
Table 9: Capability Requirements and Their Feasibility Evidence ...................................................... 17
Table 10: Evolutionary Requirements and Their Feasibility Evidence ............................................... 18
Table 11: Rationales for Selecting Architected Agile Model ............................................................... 19
Table 12: Requirement Prioritization .................................................................................................... 20
Table 13: Risk Assessment ..................................................................................................................... 21
Table 14: NDI Products Listing ............................................................................................................. 24
Table 15: NDI Evaluation ....................................................................................................................... 25
Table of Figures

Figure 1: ROI Analysis ................................................................. 13
Figure 2: ROI Analysis with Software Costs Graph ................................ 14
1. Introduction

1.1. Purpose of the FED Document

The purpose of this feasibility evidence document is to show that the processes, architecture, and plans proposed in various artifacts can be fulfilled with the time duration of this project (1 semester).

This document provides a detailed analysis of the requirements agreed upon during the Win-Win negotiations as well as the risks associated with realizing these requirements. It provides evidence to show that the system built by the end of the allocated time period, will satisfy all of the client requirements. This document indicates both completeness and consistency with all the documents presented over the duration of this course.

1.2. Status of the FED Document

There have been some changes in the client requirements after the second Win-Win negotiation session. Due to these new requirements, there have been changes in some sections of the document. It has also impacted the overall life cycle plan of the project.

- Delays caused by the client in providing UI design has led to reduction in the time available for development and impacted the resource and time feasibility of some features.
- Changes have been made to Level of Service requirements after latest meeting with the client
- More information about detailed analysis and selection of NDI have been added to section 6.2
## 2. Business Case Analysis

### Table 1: Business Case Analysis

<table>
<thead>
<tr>
<th>Assumptions:</th>
<th>Stakeholders</th>
<th>Initiatives</th>
<th>Value Propositions</th>
<th>Beneficiaries</th>
</tr>
</thead>
<tbody>
<tr>
<td>● User has a smartphone/laptop with access to internet</td>
<td>● Developers</td>
<td>● Develop the system</td>
<td>● Provide clients with a highly customizable blog</td>
<td>● Client</td>
</tr>
<tr>
<td>● User needs assistance in order to follow a healthy diet</td>
<td>● Maintainers</td>
<td>● Maintain the system</td>
<td>● Provide clients with efficient SEO to increase traffic on the blog</td>
<td>● Maintainer</td>
</tr>
<tr>
<td>● User wants to know more about his/her nutrition intake</td>
<td>● Blog Visitors</td>
<td>● Ensure efficient SEO</td>
<td>● Provide admins with a dashboard to preview, add, update, delete articles with ease.</td>
<td>● User</td>
</tr>
<tr>
<td></td>
<td>● Client</td>
<td>● Knowledge transfer to admins</td>
<td>● Allow users to like, comment on, and share blog posts.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>● Expert Dieticians</td>
<td>● Regular sync up with clients to share updates and get feedback</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Cost (Cost factors)

- Ongoing cost of personnel to maintain the system.
- Cost associated with getting the domain name
- Cost associated with AWS cloud server for hosting and auto deployment

### Benefits (Key performance indicators – KPIs)

- Increase in traffic (tracked by google analytics)
- Increase in conversions leading to more revenue
- Decrease in overhead for adding new articles
2.1. Cost Analysis

The costs associated with this project from the client’s perspective is very minimal. The hours spent by the client so far include the weekly team meetings, about 6 hours of UI/UX designer work from the Meta Nutrition team designer for the UI of the blog interface. On the software/hardware side, currently our team is using the free tier of AWS services for deployment and storage. Once the project is delivered, the blog hosting can continue on AWS as per the client’s plan of choice. This would be dependent on the kind of site traffic changes the client expects once the project is delivered.

2.1.1. Personnel Costs

For personnel costs we have included the costs for Client hours and the cost for the UI/UX designer who designed the blog layout. It also includes the time spent in maintaining the system.

<table>
<thead>
<tr>
<th>Activities</th>
<th>Time Spent (Hours)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Development Period</strong></td>
<td></td>
</tr>
<tr>
<td>Client Meeting: 1h/w * 12 w</td>
<td>12</td>
</tr>
<tr>
<td>Win-Win Sessions: 2 sessions * 1h</td>
<td>2</td>
</tr>
<tr>
<td>Training: 2h</td>
<td>2</td>
</tr>
<tr>
<td>Fill up initial blog posts into the platform and organize it: 5h</td>
<td>5</td>
</tr>
<tr>
<td>FCR ARB: 1 session * 1.5h</td>
<td>1.5</td>
</tr>
<tr>
<td>CCD: 1 session * 1h</td>
<td>1</td>
</tr>
<tr>
<td>TRR ARB: 1 session * 1.5h</td>
<td>1.5</td>
</tr>
<tr>
<td>UI/UX Design for blog interface</td>
<td>6</td>
</tr>
</tbody>
</table>
### 2.1.2. Costs associated with deployment on cloud services

Our project will not have any hardware/software related costs but will have costs associated with deployment on AWS once the development phase is over and we exhaust our free tier of AWS services. We are using open source and freely available software for development, hence there are no costs associated with software services. Hence, the development phase does not have any costs associated with cloud services due to the free tier.

**Table 3: Deployment Costs - Operation**

<table>
<thead>
<tr>
<th>Type</th>
<th>Cost</th>
<th>Rationale</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deploying on cloud services (AWS)</td>
<td>Use a free tier in the beginning for a year and then choose a pay as you go plan depending on the traffic and scale.</td>
<td>This is the approximate cost for AWS for hosting a Blog with the kind of traffic we are expecting. This cost can be reduced considerably by purchasing plans in advance and optimizing the service as per the client’s specific needs. This will depend on the client’s choices during the operation phase.</td>
</tr>
</tbody>
</table>

### 2.2. Benefit Analysis

The new system is being built from scratch and it is meant to replace the existing system. The new system provides tangible and intangible benefits. The following tables show the benefits obtained from the new system in terms of effort saved as well as the increase in visitor trends.
Table 4: Benefits of New Meta Nutrition Blog System

<table>
<thead>
<tr>
<th>Current activities &amp; resources used</th>
<th>Time taken with new system</th>
<th>% Reduce</th>
<th>Time Saved (Hours/Year)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uploading new post</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>System Admin: 2 hrs/post * 104 (2 post per week) = <strong>208 hrs/yr</strong></td>
<td>System admin: 20 mins/post * 104 = <strong>28 hrs</strong></td>
<td>83%</td>
<td>174 hrs/yr</td>
</tr>
<tr>
<td>Deploying changes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>System admin: 1 hr/deployment * 4 deployments/yr = <strong>4 hrs/yr</strong></td>
<td>15 mins/deployment * 4 deployments/yr = <strong>1 hr</strong></td>
<td>75%</td>
<td>3 hrs/yr</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td></td>
<td><strong>177 hrs/yr</strong></td>
</tr>
</tbody>
</table>

The following table describes the change in visitor trend when the new system is deployed. This is a result of efficient search engine optimization and increased customization. Increase in traffic along with better user engagement strategies leads to more conversions and thereby increases revenue for the client.

Table 5: Visitor Trend of Meta Nutrition Blog System

<table>
<thead>
<tr>
<th>Current visitor trend</th>
<th>% increase</th>
<th>Increase</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Blog viewers</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3000 viewers/month</td>
<td>33%</td>
<td>1000</td>
</tr>
<tr>
<td><strong>Conversions</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2%</td>
<td>100%</td>
<td>4%</td>
</tr>
</tbody>
</table>
Apart from some tangible benefits measured in terms of time saved, there are some intangible benefits that cannot be described as countable values. They are:

- Increased customization for the blog in terms of user interface and user experience.
- Efficient search engine optimization leading to more traffic and thereby increasing revenue.
- Increased admin capabilities such as marking posts as admin favourites, viewing drafts, etc.
- Tracking visitor demographics to target marketing campaigns, getting more subscribers, and increasing engagement.

The following table shows how the visitor conversions from the blog contribute to the annual earnings.

### 2.3. ROI Analysis

This table shows the ROI in terms of effort over the years. It has been assumed that there is a 10% increase in maintenance costs due to the 10% increase in salary each year. So, the maintenance cost of 36 hours/year will be increased by 10% annually.

<table>
<thead>
<tr>
<th>Year</th>
<th>Cost (Hours)</th>
<th>Benefit (Effort Saved)</th>
<th>Cumulative Cost</th>
<th>Cumulative Benefit</th>
<th>ROI</th>
</tr>
</thead>
<tbody>
<tr>
<td>2020</td>
<td>34</td>
<td>0</td>
<td>34</td>
<td>0</td>
<td>-1</td>
</tr>
<tr>
<td>2021</td>
<td>36</td>
<td>177</td>
<td>70</td>
<td>177</td>
<td>1.52</td>
</tr>
<tr>
<td>2022</td>
<td>40</td>
<td>177</td>
<td>110</td>
<td>354</td>
<td>2.44</td>
</tr>
<tr>
<td>2023</td>
<td>44</td>
<td>177</td>
<td>154</td>
<td>531</td>
<td>2.45</td>
</tr>
</tbody>
</table>
This table indicates the cost and benefit which can be described as:

- **Cost**
  - 31 (hours for development) + 36 (hours for maintenance) for the first year (2020)
  - Only maintenance cost for the following years with a 10% increase to show the increase in salary of employees.

- **Benefit**
  - Hours saved by the new system as compared to the old system.

The following table shows ROI analysis in terms of monetary costs. It analyses the hardware and software cost along with the personnel cost. The following table has been created by considering a $30/hour salary for personnel involved in maintaining the system.
### Table 7: ROI Analysis with software cost

<table>
<thead>
<tr>
<th>Year</th>
<th>Cost ($/year)</th>
<th>Benefit (Earnings)</th>
<th>Cumulative Cost</th>
<th>Cumulative Benefit</th>
<th>ROI</th>
</tr>
</thead>
<tbody>
<tr>
<td>2020</td>
<td>67*30 (system development) = $2190</td>
<td>0</td>
<td>$2010</td>
<td>0</td>
<td>-1</td>
</tr>
<tr>
<td>2021</td>
<td>36 * (30) = $1080</td>
<td>$2775</td>
<td>$3090</td>
<td>$2775</td>
<td>-0.10</td>
</tr>
<tr>
<td>2022</td>
<td>40 * (30) = $1380</td>
<td>$2775</td>
<td>$4470</td>
<td>$5550</td>
<td>0.24</td>
</tr>
<tr>
<td>2023</td>
<td>44 + (30) = $1500</td>
<td>$2775</td>
<td>$5970</td>
<td>$8325</td>
<td>0.39</td>
</tr>
</tbody>
</table>

**Figure 2: ROI Analysis with Software Costs Graph**

ROI Analysis with Software Costs

![ROI Analysis with Software Costs Graph](image)

Marker shows benefit realization
3. Architecture Feasibility

3.1. Level of Service Feasibility

<table>
<thead>
<tr>
<th>Level of Service Requirement</th>
<th>Product Satisfaction</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOS-1: Availability</td>
<td>Product Strategies:</td>
</tr>
<tr>
<td></td>
<td>Lightweight NDI, code</td>
</tr>
<tr>
<td></td>
<td>optimization, auto-</td>
</tr>
<tr>
<td></td>
<td>scalable cloud</td>
</tr>
<tr>
<td></td>
<td>server instances</td>
</tr>
<tr>
<td></td>
<td>Process Strategies:</td>
</tr>
<tr>
<td></td>
<td>Performance analysis,</td>
</tr>
<tr>
<td></td>
<td>prototyping, load</td>
</tr>
<tr>
<td></td>
<td>testing</td>
</tr>
<tr>
<td>Analysis: The system should</td>
<td></td>
</tr>
<tr>
<td>always be up and running</td>
<td></td>
</tr>
<tr>
<td>which implies 100% uptime</td>
<td></td>
</tr>
<tr>
<td>excluding AWS downtimes.</td>
<td></td>
</tr>
<tr>
<td>The acceptable level for</td>
<td></td>
</tr>
<tr>
<td>server uptime in the worst</td>
<td></td>
</tr>
<tr>
<td>case would be 99% every day.</td>
<td></td>
</tr>
<tr>
<td>Load testing is used to</td>
<td></td>
</tr>
<tr>
<td>ensure that the system does</td>
<td></td>
</tr>
<tr>
<td>not crash when it has to</td>
<td></td>
</tr>
<tr>
<td>process a large number of</td>
<td></td>
</tr>
<tr>
<td>simultaneous requests.</td>
<td></td>
</tr>
<tr>
<td>LOS-2: System response time</td>
<td>Product Strategies:</td>
</tr>
<tr>
<td></td>
<td>Platform feature</td>
</tr>
<tr>
<td></td>
<td>exploitation, efficient</td>
</tr>
<tr>
<td></td>
<td>AWS cloud server</td>
</tr>
<tr>
<td></td>
<td>instances, lightweight</td>
</tr>
<tr>
<td></td>
<td>backend framework</td>
</tr>
<tr>
<td></td>
<td>Process Strategies:</td>
</tr>
<tr>
<td></td>
<td>Efficient search</td>
</tr>
<tr>
<td></td>
<td>algorithms, Caching</td>
</tr>
<tr>
<td></td>
<td>of results, Prototyping, Simulation</td>
</tr>
<tr>
<td>Analysis: The system involves</td>
<td></td>
</tr>
<tr>
<td>searching for articles based</td>
<td></td>
</tr>
<tr>
<td>on keywords. This involves</td>
<td></td>
</tr>
<tr>
<td>performing search through</td>
<td></td>
</tr>
<tr>
<td>the database to get all</td>
<td></td>
</tr>
<tr>
<td>matching posts. The response</td>
<td></td>
</tr>
<tr>
<td>time should ideally be less</td>
<td></td>
</tr>
<tr>
<td>than 2 seconds but upto 4</td>
<td></td>
</tr>
<tr>
<td>seconds is acceptable.</td>
<td></td>
</tr>
<tr>
<td>LOS-3: Scalability</td>
<td>Product Strategies:</td>
</tr>
<tr>
<td></td>
<td>Using Cloud services (AWS) to scale</td>
</tr>
<tr>
<td></td>
<td>quickly and easily, Use of REST APIs</td>
</tr>
<tr>
<td></td>
<td>Process Strategies:</td>
</tr>
<tr>
<td></td>
<td>Load testing, simulating increased traffic with 10k visitors</td>
</tr>
<tr>
<td>LOS-4: Cross Platform Support</td>
<td>Product Strategies: Using responsive technology such as next.js to support cross platform use</td>
</tr>
<tr>
<td>Process Strategies: Testing across different platforms:</td>
<td></td>
</tr>
<tr>
<td>● Desktop (Chrome, Firefox, Safari)</td>
<td></td>
</tr>
<tr>
<td>● Mobile (Android, iOS)</td>
<td></td>
</tr>
<tr>
<td>● Tablet (Android, iOS)</td>
<td></td>
</tr>
</tbody>
</table>

Analysis: Since the blog is using a responsive tech stack and is hosted on the web, it can be accessed by any device size and platform respectively without requiring explicit development needs for a platform or device size.

| LOS-5: Interoperability | Product Strategies: REST API, lightweight backend |
| Process Strategies: Modular design, prototyping, simulations, integration testing, separation between client side and server side |

Analysis: System has to be built in a way that it can be evolved easily. The modular design and separation between client side and server side ensures that the system can be extended to iOS and Android apps. It also ensures ease with respect to maintaining the system.
3.2. Capability Feasibility

Most of the capabilities have been implemented by our team and they serve as feasibility evidence.

**Table 9: Capability Requirements and Their Feasibility Evidence**

<table>
<thead>
<tr>
<th>Capability Requirement</th>
<th>Product Satisfaction</th>
</tr>
</thead>
<tbody>
<tr>
<td>CR-1: Comments feature with comment sanitization</td>
<td>Software/Technology used: Regex for logic, Express for backend, React for front-end, MongoDB</td>
</tr>
<tr>
<td>Feasibility Evidence: This capability was developed and presented at our prototype presentation and its functionality was successfully proved, thus serving as a feasibility evidence. The comment box allows users to add comments and our backend checks for vulgar language.</td>
<td></td>
</tr>
<tr>
<td>Referred use case diagram: Figure 3 in SSAD document</td>
<td></td>
</tr>
<tr>
<td>CR-2: Search for posts by keyword</td>
<td>Software/Technology used: React for front-end, node.js and MongoDB on the backend</td>
</tr>
<tr>
<td>Feasibility Evidence: This capability was also developed and presented at our prototype presentation and its functionality was successfully proved, thus serving as a feasibility evidence. Users can search for posts by keyword.</td>
<td></td>
</tr>
<tr>
<td>Referred use case diagram: Figure 3 in SSAD document</td>
<td></td>
</tr>
<tr>
<td>CR-3: Auto Deployment feature</td>
<td>Software/Technology used: Github/AWS</td>
</tr>
<tr>
<td>Feasibility Evidence: This feature has been implemented and demonstrated during one of our client meetings. We showed how a commit pushed to our production branch on Github gets auto deployed on AWS.</td>
<td></td>
</tr>
<tr>
<td>Referred use case diagram: Figure 4 in SSAD document</td>
<td></td>
</tr>
<tr>
<td>CR-4: Managing posts</td>
<td>Software/Technology used: React for front-end, node.js and MongoDB for backend</td>
</tr>
</tbody>
</table>
Feasibility Evidence: We have implemented an admin dashboard that allows, adding new posts, updating and deleting existing posts.

Referred use case diagram: Figure 4 in SSAD document

<table>
<thead>
<tr>
<th>CR-5: Analytics</th>
<th>Software/Technology used: Google Analytics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feasibility Evidence: We will use Google Analytics for tracking user behaviour. Adding this functionality is fairly easy. With just a few lines of code we can add google analytics and then use its dashboard for user behaviour data.</td>
<td></td>
</tr>
<tr>
<td>Referred use case diagram: Figure 4 in SSAD document</td>
<td></td>
</tr>
</tbody>
</table>

### 3.3. Evolutionary Feasibility

The following table defines the system's evolutionary requirements and well as the steps taken to satisfy them.

**Table 10: Evolutionary Requirements and Their Feasibility Evidence**

<table>
<thead>
<tr>
<th>Evolutionary Requirement</th>
<th>Product Satisfaction</th>
</tr>
</thead>
<tbody>
<tr>
<td>ER-1: Managing user profiles</td>
<td>Software/Technology used: Node.JS, MongoDB, Next.JS</td>
</tr>
<tr>
<td>Feasibility Evidence: Creation of user profiles is a near replication of the existing admin profile. Hence it is feasible and allows the admin to track user actions.</td>
<td></td>
</tr>
<tr>
<td>Referred use case diagram: Fig 3 and Blog Visitor class in fig 6 of SSAD document</td>
<td></td>
</tr>
</tbody>
</table>

### 4. Process Feasibility
We will be using the Architected Agile model while working in biweekly cycles. In order to minimize risk, we have team members providing updates on their progress 3 times a week. In order to stay up to date with the latest developments, we have weekly meetings with the client. The table below shows reasons for following this process.

**Table 11: Rationales for Selecting Architected Agile Model**

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Rationale</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency of code integration</td>
<td>1-2 week. As per requirements of 577A and the need to prevent bugs during integrations, performing integrations more often helps our cause.</td>
</tr>
<tr>
<td>NDI Support</td>
<td>Our application is using NDI such as mongoDB, AWS EC2, Github, Node.js, Next.js. We are also using other frameworks to support additional client requirements. In order to ensure smooth development, we have ensured that these NDIs are compatible with the system architecture.</td>
</tr>
<tr>
<td>Personnel experience</td>
<td>Team members have previous experience of working with an agile model</td>
</tr>
<tr>
<td>Project size</td>
<td>Medium size project with incremental updates</td>
</tr>
<tr>
<td>Key Stage I Activities : Incremental Definition</td>
<td>Valuation/Architecture Commitment Reviews</td>
</tr>
<tr>
<td>Time per Build; Time per Increment</td>
<td>Time/Build = 1-2 weeks – This complies with the characteristics of 577A class, which recommends that the team come up with lots of builds during the development. Time/Increment = 1-2 months – This also complies with 577A class, which allows the team to incrementally implement the system within 6 months.</td>
</tr>
</tbody>
</table>
Since this is a one semester project, the team is following schedule based reviews. By following agile development and having frequent meetings with the client, the team is making sure that all artifacts are delivered as per schedule. In order to stick to schedule and ensure that the requirements are feasible, the stakeholders are incrementally committing to the requirements.

Here is a table which shows how requirements have been prioritized.

<table>
<thead>
<tr>
<th>Priority</th>
<th>Requirements</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Auto deployment of code to AWS EC2</td>
<td>CR-3</td>
</tr>
<tr>
<td>2</td>
<td>Search for posts by keyword</td>
<td>CR-2</td>
</tr>
<tr>
<td>3</td>
<td>Managing posts (for admin)</td>
<td>CR-4</td>
</tr>
<tr>
<td>4</td>
<td>Comment Sanitization</td>
<td>CR-1</td>
</tr>
</tbody>
</table>

All the team members have been assigned different roles for creating multiple deliverables. Trello is being used for task management and to track the progress of each team member. We have used COCOMO II for estimating the cost and effort for the project.
5. Risk Assessment

Table 13: Risk Assessment

<p>| Risk Description                                    | P(L) 1-10 | S(L) 1-10 | RE | Risk Mitigation Action Items                                                                 |
|-----------------------------------------------------|-----------|-----------|----|============================================================================================|
| 1. Additional Client Requirements                   | 7         | 7         | 49 | Determine the feasibility of the new requirements and prioritize them. Also ensure that the high priority requirements are not affected by these newly introduced requirements. |
| Last week the client introduced new requirements for the admin side dashboard where they want to view previews of a blog post before publishing and also mark posts as favorites. These requirements are completely new. This causes a risk to the existing timeline and may lead to the project not being completed within the allocated time period. |
| 2. Inefficient SEO on the new website                | 6         | 8         | 48 | Buying information by discussing with the client if he has an SEO expert for us to work in collaboration with or provide an alternative solution. SEO testing and quantification will take place after creating and deploying the new website. |
| One of the main reasons for creating a new website is to have efficient SEO. Since no one in the team has experience in improving SEO, it may be possible that the new website would not deliver the required level of SEO. This would impact the traffic on the blog and thereby impact the business as well. |</p>
<table>
<thead>
<tr>
<th></th>
<th><strong>3. Incomplete client-side UI design for search page layout</strong></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Client has provided UI mockups for some web pages. But UI mockups for the search page have not been provided. Since searching for articles is an important user</td>
<td>4</td>
<td>5</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>Follow up with the client with a UI mockup for the search layout and start developing this layout based on their feedback.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th><strong>4. Constraints imposed by client on technology to be used</strong></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>As per the most recent meeting, the client wants us to use SCSS for styling and have a highly customizable text editor (open source editor for React.JS) for uploading new articles. Hence we need to research on whether or not it is compatible with our application.</td>
<td>3</td>
<td>6</td>
<td>18</td>
</tr>
<tr>
<td></td>
<td>Perform research about the use of SCSS and find available COTS products that can be used to create a customizable editor and are compatible with this application</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th><strong>5. Redesign UI for admin panel</strong></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The client was not happy with the Admin UI designed by the team and wants us to redesign the entire UI to match his vision which has not been clearly stated by him.</td>
<td>3</td>
<td>5</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>We can refer to the UI designs for the customer side sent to us and create a similar UI feel for the admin and send a few wireframes to the client to review before development.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th><strong>6. Blockers arising due to lower frequency of meetings with clients</strong></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>In some situations the client is busy to attend meetings or has little time for updates and review meetings which has</td>
<td>2</td>
<td>4</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>We can set up a review channel by asking the client to send his feedback for each important decision if he is unable to attend the review meetings</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
slowed us down a bit

<table>
<thead>
<tr>
<th>7. Performance may be compromised while searching articles</th>
<th>1</th>
<th>3</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>The client requires a feature to search for articles using tags and titles based on the keyword entered by the user. Since this involves searching through text, it may impact the response time and increase latency when the number of blog articles increase.</td>
<td>We need to ensure that our prototype for search can perform efficiently and quickly with a large number of articles as well. In order to do this, we have performed testing with large number of posts and multiple search requests</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
6. NDI/NCS Interoperability Analysis

6.1. Introduction
To keep the development tools cost low, we decided to use popular open source software and libraries that are continuously updated and supported and are compatible with each other.

6.1.1. COTS / GOTS / ROTS / Open Source / NCS
The table describes the products used in our project

<table>
<thead>
<tr>
<th>NDI/NCS Products</th>
<th>Purposes</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS</td>
<td>Cloud computing, hosting and deployment</td>
</tr>
<tr>
<td>MongoDB</td>
<td>Cross-platform, document-oriented database program</td>
</tr>
<tr>
<td>Next.js</td>
<td>React framework for the front end</td>
</tr>
<tr>
<td>Node.js</td>
<td>Open Source JS server environment</td>
</tr>
<tr>
<td>Express</td>
<td>Backend Web Application framework for node.js</td>
</tr>
<tr>
<td>Rich Markdown Editor</td>
<td>Creating a customizable markdown editor in Next.JS</td>
</tr>
</tbody>
</table>

6.1.2. Connectors
In this project we use the cloud framework of MongoDB which allows our AWS deployed application to retrieve data from our MongoDB database. We have also used the mongoose library of Node.JS to access the database using node.JS. We are using Express library in Node.JS to create REST APIs that will be used by the client side of our website.
6.1.3. Legacy System

The legacy system is based on the Wix platform which will be retired once this project is in use. Thus, there is no need to maintain compatibility with the legacy system.

6.2. Evaluation Summary

Table 15: NDI Evaluation

<table>
<thead>
<tr>
<th>NDI</th>
<th>Usages</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS</td>
<td>Cloud computing services and deployment</td>
<td><strong>Positive points</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Cheaper than other cloud computing platforms</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Widely used</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Documentations available</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Easy to scale</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Easy to develop/deploy on</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Development team familiarity</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Negative points</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- The only cost component in our software choices</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Monthly costs can increase without careful management</td>
</tr>
<tr>
<td>MongoDB</td>
<td>Database</td>
<td><strong>Positive points</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Free to use</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Documentations available</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Easy to scale</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Easy to develop with</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Cross platform availability</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Document DB (Schema less)</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Negative points</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Not a relational DB</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Usually slower than other more simpler DB choices such as MySQL</td>
</tr>
<tr>
<td>Next.js</td>
<td>Front-end Framework</td>
<td><strong>Positive points</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Automatic server rendering and code splitting</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Built with React</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Easy setup</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Documentations available</td>
</tr>
<tr>
<td>Tool</td>
<td>Description</td>
<td>Positive points</td>
</tr>
<tr>
<td>----------------</td>
<td>------------------------------</td>
<td>--------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Node.js</td>
<td>JS server environment</td>
<td>- Free to use</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Widely used</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Documentations available</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Easy to develop/deploy on</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Development team familiarity</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Largest source of open source libraries</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Open source and widely supported</td>
</tr>
<tr>
<td>Express</td>
<td>Backend framework</td>
<td>- Free to use</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Widely used</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Documentations available</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Easy to develop/deploy on</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Development team familiarity</td>
</tr>
<tr>
<td>Rich Markdown</td>
<td>Markdown Text Editor</td>
<td>- Free to use</td>
</tr>
<tr>
<td>Editor</td>
<td></td>
<td>- Easy to use on webpage</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Highly customizable</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Availability of support</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Not very simple for developer</td>
</tr>
</tbody>
</table>