EJSIT
European Journal of Science, Innovation and Technology
ISSN: 2786-4936
www.ejsit-journal.com
Volume 5 | Number 2 | 2025

Designing AI-Powered Website Builder Using the DSDM Framework

Sana Rizwan[1], Hafiz Faizan[2], Muhammad Zain[3], Fasih Ur Rehman[4], and Rehan Rabbani Baig[5]
[1]Assistant Professor, Department of Computer Science,
COMSATS University Islamabad, Lahore Campus, Pakistan
[2,3,4]Department of Computer Science,
COMSATS University Islamabad, Lahore Campus, Pakistan
[5]University of Engineering and Technology Lahore, Pakistan

ABSTRACT

In contemporary software challenges, budget control and time management are the critical features while handling software construction and managerial issues. These problems can optimize and control in better ways with the help of collaboration and communication with customers and other stakeholders. Eventually software construction is an iterative process so agile project management activities will be the superior choice to deal with challenges. Dynamic Systems Development Method (DSDM) is an agile framework that is designed according to the business perspective, business needs, project sizes and demands by delivering real value early in the project lifecycle. This era is working with AI-powered software applications, DSDM seems a smart solution to cope with difficult integrating technologies. DSDM practices like timeboxing, MoSCoW prioritization, prototyping, facilitated workshops, help in organizing development, focus on essential tasks, and maintain effective teamwork. These methods focus on step-by-step progress and align working with evolving AI requirements. AI related challenges can be managed by balancing structured development approaches. These kinds of approaches help in building sustainable and reliable systems while open to changes. In this paper, we are trying to implement and make decisions using DSDM agile framework in the development of AI based software applications.

Keywords: Dynamic Systems Development Method (DSDM), Agile Framework, MoSCoW, Agile Manifesto, Adaptive planning, Cross-functional collaboration

INTRODUCTION

AI is changing the development of software systems by making systems that help in automating complex tasks. There are some challenges in making AI related Projects like unpredictable results, evolving requirements, and the need for continuous refinement. These kinds of challenges overcome through Agile framework like DSDM. We as a team organize our tasks, adapt to incoming user requirements, improve efficiency and productivity using some techniques of DSDM like timeboxing, MoSCoW prioritization, prototyping, facilitated workshops. These methods help in breaking complex AI processes ensuring enough time to do experimentation in development. We can follow a streamlined workflow using this method. In Order to evaluates e DSDM's effectiveness in development we conduct observations, team surveys, and project documentation analysis. Our focus was to understand how DSDM helps in planning, execution of AI software and at same time handle challenges like uncertain model performance, data driven changes. The following sections show a detailed analysis of our approach, key findings and lessons learned. We believe this research helps in understanding how DSDM helps to build software solutions that are more structured, reliable and result oriented.

Agile software was created to help in solving challenges of traditional development because it is not flexible to changes. The Agile Manifesto (2001) provides key principles that prioritize functional development, customer collaboration and adaptiveness over rigid planning. The popular Agile frameworks like DSDM (Stapleton, 2003) follow structured workflows that align with business needs and ensure continuous user involvement, clear project goals, and iterative deliveries. It builds on top of concepts like fitness for business purposes, iterative cycles having defined deadlines, and active collaboration. Agile methodologies help to boost success rates, accelerating delivery, and increasing customer satisfaction. They focus on customer satisfaction, collaborative teamwork and flexible to changes.

Applying Agile in AI and Web Development Projects

By Using Agile practices in AI and web development (Stapleton, 2003) we have both opportunities and challenges. Case studies highlight how Agile frameworks can adapt to these hard fields:

- A machine learning fraud detection project uses DSDM (Smith, 2020) to manage dynamic requirements. This helps in team collaboration, iterative cycles, and adaptive planning to address shifting data and priorities.
- A survey of AI practitioners showing how Agile methods like regular feedback, cross-functional teamwork, and iterative development help in gaining effectiveness for AI projects.

DSDM helps with rapid development of web-based platforms by making users more engaging to the system. However, some challenges might arise.

Research Gaps and Opportunities

There are some areas where research is needed still like:

- Agile for AI-Driven Web Apps: Agile is not commonly used got AI-based web application, so this study is about to fill that gap by examining real-world use.
- Cross-Functional Collaboration: AI projects need seamless teamwork between data scientists, developers, and domain experts. This study explores how frameworks like DSDM helps for doing this collaboration and save time.

In rapidly changing fields of AI web development, we can take advantage from agile methods.

MATERIALS AND METHODS

We decided to use DSDM because it offers a **structured yet flexible** approach to software development. It provided a clear framework for managing our tasks while allowing us to adapt to changes as needed. Here's why DSDM was the best choice for our project:

- **Iterative Development:** Breaking the work into smaller steps allowed us to continuously improve the system.
- **Clear Prioritization:** The **MoSCoW prioritization method** helped us focus on the most important features first.
- **Timeboxing:** We worked within fixed timeframes to stay on schedule and avoid delays.
- **Stakeholder Involvement:** Regular feedback ensured that we built something truly useful.
- **Fitness-for-Purpose Approach:** AI model performance was closely monitored to ensure quality.

Methodology

We analyzed how DSDM framework was applied in AI powered web app solutions. During development we face some challenges, so we make the decision to implement agile practices. We followed an interpretive approach, meaning we focused on understanding the project from the perspective of the team members. Our aim is to gather information on how agile methodologies adopted to unexpected changes, and how they ensured that AI features worked as intended. We have collected data through direct observations, team discussions, and project documentation, which help us to understand about these processes.

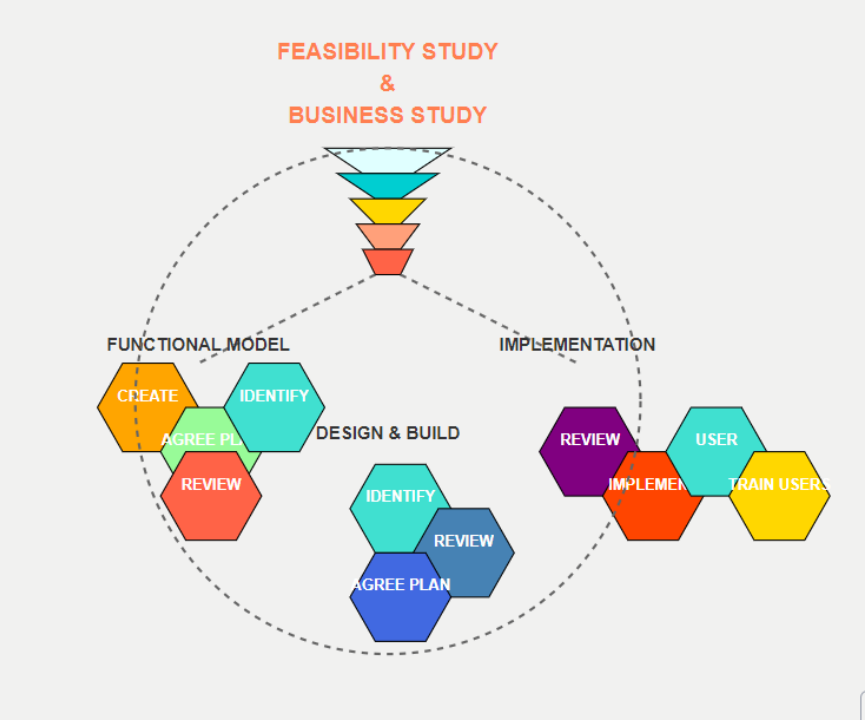

Figure 1

Figure 1 shows the development process of the AI project using the DSDM approach. Before starting the development, the project goes through the **Pre-Project** phase. It is basically the planning phase. Instead of just starting on development, first, the project is checked. Is it worth it? What problem will it solve? Project goals and stakeholders are identified, and decisions are made to start working on the project or not. The **Feasibility Study** ensures whether the project is practically possible or not. Risks are identified that can cause obstacles on the road to project success. Do we have the required team, budget, and resources? Is the project requirement mentioned by the stakeholder possible to achieve or not? After the project goes through the feasibility stage, a **Business Study** is conducted. A detailed analysis is conducted on the project. Objectives are defined clearly. Requirements are gathered using one of the DSDM techniques (MoSCoW Prioritization (MoSCoW Prioritization Explained, 2024)) to sort the requirements so that the most important ones are first to be done. All the resources that are needed and the technical requirements of the staff are identified.

Before the development team starts working on the assigned features of the project, in the **Functional Model Iteration** phase, prototypes of those functions are created so that they can be tested. Requirements are clarified. Each requirement model is built incrementally. In every phase, user feedback is continuously gathered. In the **Design and Build Phase**, the actual development starts after the functional model is done. The development team follows

the DSDM Timeboxing technique (Timeboxing in DSDM, 2024), where the task and time are defined, making the development in a structured way. Work is deployed in small iterations. In each iteration, functionality is decided depending on the top priority, then designed and coded, and in the end, it is deployed. This cycle is followed for each iteration. In the **Implementation Phase**, the whole project is deployed into production, as in the previous phase, we worked on small iterations, but here the whole product is deployed live, making it operational for launch. This is one of the final phases of the project. Documentation and reviews are done. All the guidelines of the project are assembled, and the product is reviewed to ensure it meets the criteria for which it was made, and in the end, it is deployed in the market, making it live to use.

The **Post Project** phase is the last phase in the DSDM process. As the project is now deployed. Here comes its maintenance. Different ways are used daily/weekly feedback is gathered from the customers, looking for any issues that may have arisen. Performance is monitored to find bugs and check the areas where optimization is required. If needed, then plans are created for the next update.

Key DSDM Practices

1. Timeboxing

As shown in Table 1, Timeboxing played a key role in keeping our project on track. We divided the work into clear phases with fixed durations to ensure steady progress. Below is how we structured our timeline:

Table 1

Phase	Tasks	Duration
Planning	Gathering requirements, discussing design ideas	2 Weeks
Development	Building AI and website features in iterations	6 Weeks
Testing & Review	Fixing bugs, running user tests, gathering feedback	3 Weeks
Deployment	Finalizing everything and ensuring security	2 Weeks

2. MoSCoW Prioritization

To manage our workload effectively, we used the MoSCoW method to separate essential features from optional ones. In Table 2 we categorized our tasks:

Table 2

Priority	Features Implemented
Must-Have	Secure authentication, NLP-powered dashboard, security measures
Should-Have	Browser compatible, deployment automation
Could-Have	Custom UI themes, advanced analytics
Won't-Have	AR/VR features (considered for future versions)

This method made it easier for us to focus on delivering the most important aspects first while keeping additional ideas in mind for later.

3. Workshop Facilitation

Workshops were essential in keeping everyone on the same page. We organized three main types of workshops to ensure smooth collaboration as shown in Table 3.

Table 3

Workshop Type	Purpose	Participants
Planning Workshop	Define project goals, gather requirements	Developers, Managers
Review Workshop	Evaluate completed work, gather feedback	Developers, QA Team
Deployment Workshop	Ensure system stability and security	DevOps, Security Team

These sessions helped us identify potential issues early, make necessary adjustments, and ensure that the project stayed aligned with our goals.

Approach Adapted DSDM for AI Development

Since AI-powered projects require additional flexibility, we made a few modifications to the standard DSDM framework:

- **Extended timeboxes to 4 weeks** per iteration to allow enough time for AI model training and testing.
- **Stronger collaboration between AI/ML Teams and developers** to ensure smooth AI integration.
- **Defined AI-specific success criteria**, such as accuracy, speed, and reliability.
- **Automated testing and deployment pipelines** to speed up development and catch errors early.

Our methodology combined DSDM's structured approach with AI-specific adaptations to ensure smooth and efficient development. Timeboxing kept us on track, MoSCoW prioritization helped us focus on what mattered most, and workshops improved collaboration. This combination helped us balance the flexibility needed for AI with the structured workflow required to meet deadlines and deliver a high-quality product.

Implementation of DSDM Techniques

1. Timeboxing

Timeboxing is done in project to achieve the milestone within a specific time. To create Timeboxing, Project is divided in smaller chunks which will be the timeboxes. Objectives are set for each timebox. In each Timebox, tasks are sorted in a way that highest priority task is done first. Each team member is required to complete the tasks within the mentioned timeline.

Timeboxing helps in completing the task of AI Website project with the allocated time, as development team already knows what they need to do. Each key feature of the AI website, such as AI-powered recommendations, chatbot functionalities, and personalization, Security measures were broken into tasks, and specific time limits were set for the development of each one of them. Due to priority wise tasks, most important tasks of AI Website were done first, as shown in Figure 2.

Timebox	Timeframe	Focus	Weeks	Assigned To	Status	Notes
TB 1	Nov-Dec 2024	Set up dev environment, configure cloud infrastructure. Develop core AI feature API endpoints .	1-4	DevOps, AI/ML, Back-End Dev	To Do	Ensure cloud provider setup , define initial API's.
TB 2	Nov-Dec 2024	Integrate AI models. Front-end : Basic layout and responsiveness (desktop) Implement secure authentication	5-8	AI/ML, Front-End Dev, Back-End Dev	To Do	Verify model compatibility, focus on core layout, choose authentication provider.
TB 3	Nov-Dec 2024	Front-end: Mobile responsiveness. Implement basic security measures. testing of AI feature integration. Performance testing .	9-12	Front-End Dev, Back-End Dev, Tester, DevOps	To Do	Test on various devices
TB 4	Jan-Feb 2025	Browser compatibility testing Implement user customization options (e.g., themes, layouts).	13-14	Tester, Dev Team, Front-End Dev	To Do	Target specific browsers and versions.
TB 5	Jan-Feb 2025	Implement multilingual support (content translation, UI localization).	15-16	Dev Team	To Do	Research translation services/APIs.
TB 6	Mar-Apr 2025	Implement advanced user customization Implement monitoring of AI chatbot performance .	17-20	Front-End Dev, DevOps, AI/ML	To Do	Define key performance indicators for chatbot.
TB 7	May-Jun 2025	Implement AI-powered image quality improvement. personalized recommendations.	21-24	AI/ML Engineer , Dev Team	To Do	Research and select appropriate AI model. .
TB 8	Jul-Aug 2025	AI Enhancements and Personalization	25-28	Dev Team, AI/ML	To Do	Focus on core personalization features.
TB 9	Sep-Oct 2025	Implement voice command interaction with the website	29-32	Front-End Dev	To Do	Research voice recognition APIs.
TB 10	Nov-Dec 2025	Implement face login authentication. Integration and testing of all AI enhancements.	33-36	Back-End Dev, Tester	To Do	Research facial recognition APIs.
TB 11	Jan-Feb 2026	Reporting and Analytics / Integrations	37-40	Dev Team, DevOps	To Do	Define key metrics for reporting.
TB 12	Mar-Apr 2026	Develop custom reports based on user feedback.	41-44	Dev Team	To Do	Dev Team,

Figure 2

2. Workshop facilitation

We need one person who will be responsible for facilitating the workshop known as the workshop facilitator. One of the responsibilities of the facilitator is to prepare for the workshop. We identify the participants who will have to attend that workshop. Resources are identified that are needed to conduct the workshop and to keep participants acknowledge, documents are sent to them so they know what is to be done in the workshop and they should be prepared.

During the workshop all the participants' opinions are gathered to find the solution to the problem. Workshop Facilitation leads to team spirit building as everyone is involved in

solving the problem. Everyone is aware of what decisions are made, as everyone is involved. It helps in brainstorming and finding the solution to the problem faster.

The workshop was held to prioritize the AI website's core functionalities, including real-time data processing, user personalization, and chatbot accuracy. We created a plan for the project. We defined scope, technical requirements that are needed for the success of AI Website and the Core AI features that are most needed are prioritized first so they are developed earlier and a plan on how the development will execute. The selected team consists of developers, designers and stakeholders. Participants should know the required developments languages (React JS, Tailwind CSS, SQL, ML) required to build the AI powered Websites, website design principles, should have problem solving skills. Multiple facilitated workshops with developers, designers, AI experts, and business stakeholders were conducted for the AI Website Project below Figure 3 & 4.

Field	Details
Workshop Name	AI Website Development - Sprint Planning
Date	30th January 2025
Time	10:00 AM - 1:00 PM
Location	Zoom (Remote)
Facilitator	Project Manager
Participants	AI, Backend, UI/UX, Data, Tech Lead, DevOps, Security, NLP, Admin, Security Team
Objective	Define user stories, prioritize AI features, and plan tasks for development.

Figure 3

ID	Item Type	Description	Priority	Owner	Notes	Column1
1	User Story	As a user, I want to interact with an AI chatbot for support.	Must Have	AI Team	Use pre-trained AI model.	Research AI model options.
2	User Story	As a user, I want to upload images for AI-based analysis.	Should Have	Backend Team	Needs cloud storage setup.	Set up cloud storage.
3	Requirement	The website must have a responsive and user-friendly UI.	Must Have	UI/UX Team	Use modern frontend framework.	Finalize UI design.
4	Risk	Delay in AI model training due to insufficient data.	High	Data Team	Need to gather more training data.	Schedule data collection.
5	Decision	Use React.js & Node.js for frontend and backend.	-	Tech Lead	Approved by stakeholders.	-
6	Task	Set up cloud infrastructure for hosting AI models.	Must Have	DevOps Team	Needs AWS/GCP setup.	Configure cloud services.
7	User Story	As an admin, I want to monitor AI chatbot performance.	Should Have	Admin Team	Add logging and analytics.	Implement monitoring tools.
8	Requirement	The system should support multi-language responses.	Could Have	NLP Team	Requires language model setup.	Research multi-language APIs.
9	Risk	Security vulnerabilities in user data storage.	High	Security Team	Conduct security audit.	Implement data encryption.
10	Task	Conduct usability testing with early users.	Must Have	QA Team	Requires real-user feedback.	Prepare usability test cases.

Figure 4

3. MoSCoW Prioritization

It helps in prioritizing the requirements of projects. To do Moscow prioritization, first requirements of the project are gathered and then it divides the requirement into four categories which must have, should have, could have, won't have. There must have been those functionalities which are compulsory for projects to be successful without those projects will be a failure. Should have been those that are important to project but are not as critical as compared to must have. There could have been ones that are not necessary, but it would be nice to have them, they are included if we have time and resources available. In won't have, we mention the features/activities that will be not in the project so to avoid expectations.

MoSCoW Prioritization helps in developing the most valuable functionality first, so the most important thing must be completed first. It defines what to do in a structure way resulting in less confusion and making a clear roadmap for the team. Before starting the project, stakeholder requirements are clearly defined. By developing must-have features first, we can reduce the risk of project failure.

Must Have:

- The website must be both Desktop and Mobile Responsive.
- The website must support Core AI Feature functionality.
- Hosting of AI Models on Cloud Infrastructure.
- The website must implement security measures (secure authentication, defense again common attacks)

Must Not Have:

- Must not support outdated browsers
- Must not integrate with the system that are unreliable
- Must not have Hard to use Interface

Should Have:

- Should be browser compatible.
- Should allow user to further customize according to their ease of use.
- Support Multi Languages for different region of users.
- Monitoring of AI chatbots Performances

Should Not Have:

- Complex features that add minimum value.
- Avoid Dependency on technologies that are depreciated.
- Features that can cause too much maintenance and load

Could Have:

- AI Powered Image Quality Improvement.
- Give user personalized experience by using AI insights
- Interaction with website using voice command.
- Providing different ways to authenticate including Face Login.

Could Not Have:

- Updates that are not needed.
- Integration with unstable API's.
- Automatic Generation of Quality Content.
- AI Model Perfectly Understanding each user query.

Implementation

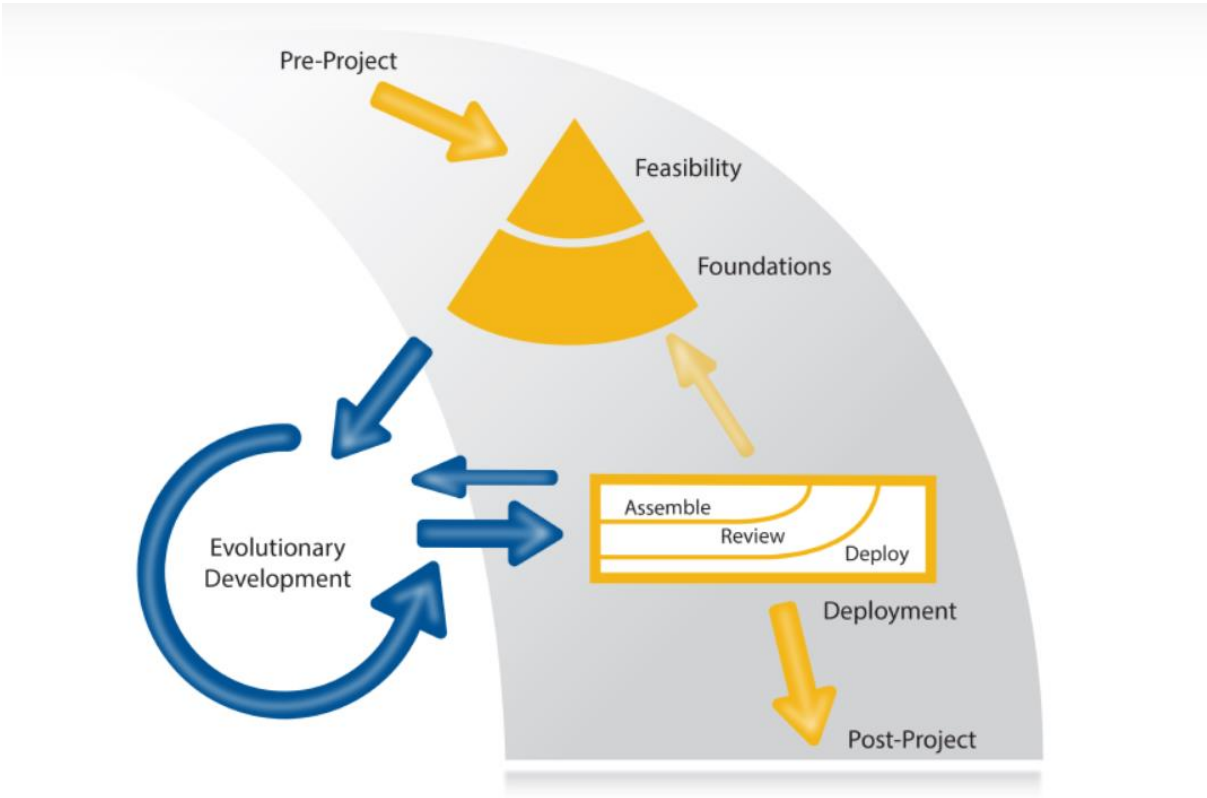

Figure 5

Ensuring that the AI Website Project meets the goals and is worth continuing. Business owners, developers, UI/UX Designers, AI Experts, Team lead were identified as stakeholders for the AI project. The goal was to build a website with Core AI Features integrated into it (AI chatbot, recommendation engine etc). Possibility was checked whether AI Core features can be developed within the timeframe and budget. If it meets the criteria, we move to the next phase (feasibility study). In the feasibility study, check out all the core functionalities of the AI Website Development were identified. Budget and Time were checked. Can we complete the AI Website Development project within the given constraints or not? Technical Requirements were checked of building AI models that will be integrated with the website.

In AI Website Development Project Business owners, developers, UI/UX Designers, AI Experts, Team lead are identified as stakeholders. All the Requirements are prioritized using DSDM technique (MoSCoW Prioritization) so the critical features of the AI are done first. Must have features like AI Chatbot, Responsiveness, Security were identified and prioritized (MoSCoW Prioritization) And Team should know programming languages that are needed. Deciding whether to use a cloud-based AI platform or develop custom AI models. Team created a detailed roadmap for the development process to execute smoothly.

Functional Prototype of the AI Website project was built and test iteratively and testing team also looks for any gaps. Teams get their respective task with assigned timeline. A basic chatbot model was built using NLP (Natural Language Processing) techniques to handle common user queries. early testing showed that the chatbot needed better contextual understanding, leading to improvement in the NLP model. AI Website development is done incrementally. Team follows the Timebox created with each AI functionality mentioned that needs to be developed within a time frame. The team starts working on building the AI Core features depending on priority and each iteration is tested and deployed. The AI driven

recommendation system was tested with real user data, adjusting the algorithm for more relevant content suggestions. After the features of AI Website are built mentioned in the timebox than the AI Website is ready to be deployed into the market. First, it is checked thoroughly it meets the criteria. It is performing all the AI tasks that were meant to be performed. After that it is deployed on one of the cloud platforms. The AI website was moved from staging server to a live production server with minimum downtime. After deployment, user behavior was tracked to further refined AI-driven recommendations. Users feedback is collected on the website to check if any AI feature is lagging and not working properly. Maintenance is done on a weekly basis to ensure the website works properly without going down. Criteria are set to check the accuracy of the AI models. For the AI website, an AI performance dashboard was implemented to monitor chatbot interactions, and response times. This helped in identifying areas for further AI model tuning. If the issue arises, that AI Feature is added in the next update plan.

RESULTS AND DISCUSSION

User Satisfaction Improvement

DSDM's way of working, with its short cycles and focus on users, helped make the AI website better. By getting feedback from the users as shown in Table 4 and making those changes, teams fixed any problems early on. As AI features were designed based on real user needs, making the AI powered website easier and more enjoyable to use.

Table 4

Metrics	Inputs	Calculation Method	Target	Result
USatL (User Satisfaction Level)	SURSQ = 420, TNRSQ = 48	USatL = 420 / 48 = 8.75	> 8.0	≥ 8.5
USatC (User Satisfaction Change)	USatLP2 = 8.75, USatLP1 = 7.8	USatC = (8.75 - 7.8) / 7.8 = 0.95 / 7.8 = 0.1218 (approximately 12.2%)	>10%	≥ 15%

Timeboxing & Delivery Efficiency

This approach was helpful for keeping us on schedule and preventing things from being delayed. The development of AI project was broken into small modules which allowed us to meet our deadlines and stay efficient. We were able to refine our AI model due to a clear timeline which gave us flexibility and as a result we got improved results in each iteration. As the result is shown in Table 5.

Table 5

Metric	Calculation Method	Target	Result
Task Completion Rate	% of tasks completed in timebox	≥ 90%	93%
Average Iteration Time	Time taken per development cycle	≤ 4 weeks	3.5 weeks
On-Time Delivery	% of deadlines met	≥ 95%	97%

AI Website Performance Optimization

As shown in Table 6, we prioritized performance from the start of this project. Using the DSDM methodology, we consistently tested the website's speed and responsiveness during each development cycle. This allowed us to catch any issues early on and address them. We also carefully fine-tuned the AI-powered features, like personalized recommendations and chatbot responses, to minimize processing time and boost overall efficiency.

Table 6

Metric	Calculation Method	Target	Result
Website Load Time	Time taken to load the homepage	≤ 2.5s	2.1s
Server Response Time	Time taken by server to respond	≤ 1.2s	1.0s

AI Model Accuracy Improvement

We have done different testing on AI model to improve its accuracy on the website. In each development cycle training data was gathered on which the AI model was trained continuously improving algo's and improving its results. Due to this AI model was able to improve its decision on new data as well as shown in Table 7.

Table 7

Metric	Calculation Method	Target	Result
AI Model Accuracy	% of correct predictions	≥ 90%	91%
Error Rate	% of incorrect predictions	≤ 10%	9%
Training Data Expansion	% increase in dataset size	≥ 20%	25%
AI Learning Speed	Time to adapt to new data	≤ 1 day	0.8 days

Deployment & Flexibility

Due to DSDM's approach, we were able to meet the changing project requirements. Table 8 states that the website remained stable, and it is easy to accept new changes without causing any issues. New features were easily added to new deployments. Testing before each deployment was done to ensure all issues are fixed and to identify if there is any bug.

Table 8

Metric	Calculation Method	Target	Result
Deployment Frequency	No. of releases per quarter	≥ 3	4
Rollback Incidents	No. of failed releases	≤ 1	0
Flexibility Score	Ability to adapt to changes (1-10)	≥ 8	9

CONCLUSIONS

Agile approach DSDM (Dynamic Systems Development Method) methodology was implemented to complete the AI Website project. DSDM offered benefits such as collaboration, fast delivery of the functional software, improving quality, and easily accepting changing requirements.

Use of DSDM Techniques helped in improving the overall quality of the AI Website project. Timeboxing helped by ensuring that the development cycle remains on track, all gathered AI Website requirements are built within the allocated time, preventing delays and providing a structured approach for team to work. Facilitation Workshop helped in gathering the AI Website requirement clearly, resolving issues and making decisions quickly without compromising on the quality. Due to effective collaboration among stakeholders, misunderstandings were minimized. MoSCoW prioritization helped in project success by prioritizing the AI Core features that are the most critical to be first one to be built. Team effectively managed the project scope by focusing on core features while keeping lower-priority tasks flexible. This ensured that critical functionalities are built on time. This also led to better resource allocation. This approach ensured that the team was able to meet the unique

demands of AI development. Continuous collaboration, workshops, prototyping, and regular feedback loops helping ensure alignment among all stakeholders.

Overall, by integrating Timeboxing, MoSCoW prioritization, Workshops, and Facilitation, and implementing the process, DSDM provided a structured yet adaptable framework that enhanced productivity, improved software stability, and delivered a high-quality AI-powered website aligned with business and user needs.

ACKNOWLEDGEMENTS

We are thankful to the Department of Computer Science, COMSATS University Islamabad Lahore Campus for their cooperation and kind support throughout our research period. Special thanks to other fellows for the guidance and support for data analysis, peer reviews, beta testing procedure, quality assurance and technical assistance.

REFERENCES

Agile Manifesto. (2001). *Principles Behind the Agile Manifesto*. Retrieved from https://agilemanifesto.org/principles.html.

MoSCoW Prioritization Explained. (2024). In *DSDM Project Framework* (Chapter 14). Agile Business Consortium. Retrieved from https://www.agilebusiness.org/dsdm-project-framework/moscow-prioririsation.html.

Smith, B. (2020). Agile in Artificial Intelligence Projects.

Stapleton, J. (2003). *DSDM: Dynamic Systems Development Method*.

Timeboxing in DSDM. (2024). In *DSDM Project Framework* (Chapter 13). Agile Business Consortium. Retrieved from https://www.agilebusiness.org/dsdm-project-framework/13-timeboxing.html.