



Green University of Bangladesh

Department of Computer Science and Engineering (CSE)
Semester: (Spring, Year: 2024), B.Sc. in CSE (Day)

Brain Station 23

KSA Report

Course Title: *Information System and Design*

Course Code: *CSE 403, Section: 212-D3*

Students Details

Name	ID
Md. Hasibur Rahman	212902018
Mohammad Arif Hossain	212902057
Sajid Rahman Rifan	212902017
Md. Yazuddin	211902046
Md. Ashaduzzaman Ashik	212902006

Submission Date: *23-05-2024*

Course Teacher's Name: *Ms. Meherunnesa Tania*

[For teachers use only: **Don't write anything inside this box**]

<u>Report Status</u>	
Marks:	Signature:
Comments:	Date:

Contents

1	Company Details	4
1.1	Company Details:	4
1.2	Services Offered:	4
1.3	Notable Global Partners:	5
2	Specific Product / Software Selection	6
2.1	Introduction:	6
2.2	Software Overview:	6
2.3	Live Preview	6
2.4	Solution:	8
2.5	Description:	8
2.6	Impact:	8
2.7	Features:	8
2.8	Limitations:	9
3	Company Employee Details	9
3.1	Executive Leadership:	9
3.2	Engineering and Development:	9
3.3	Product Management:	10
3.4	Design:	10
3.5	Data and Analytics:	10
3.6	Support and Operations:	11
3.7	Sales and Marketing:	11
3.8	Human Resources and Administration:	11
3.9	Finance and Accounting:	11
3.10	Project Management:	11
3.11	Research and Development:	12
3.12	Training and Development:	12
3.13	Content Creation:	12
4	Selected Product Model SDLC	12
4.1	Why Agile SDLC is Used:	12
4.2	12 Agile principles:	12

5	Total Phase Completion	14
5.1	5 Phases	14
5.2	Gantt Chart	15
6	Planning	15
6.1	Objectives:	15
6.2	Activities:	15
6.3	Deliverables:	16
7	Analysis	16
7.1	Objectives:	16
7.2	Activities:	16
7.3	Deliverables:	17
8	Design	17
8.1	Objectives:	17
8.2	Activities:	17
8.3	Deliverables:	17
9	Implementation	18
9.1	Implementation Phase:	18
9.2	Objectives:	18
9.3	Activities:	18
9.4	Deliverables:	19
10	Maintenance	19
10.1	Four major activities:	19
10.2	Obtaining maintenance requests	19
10.2.1	Objective:	19
10.2.2	Activities:	19
10.2.3	Deliverables:	20
10.3	Transforming requests into changes	20
10.3.1	Objective:	20
10.3.2	Activities:	20
10.3.3	Deliverables:	20
10.4	Designing changes	21

10.4.1 Objective:	21
10.4.2 Activities:	21
10.4.3 Deliverables:	21
10.5 Implementing changes	21
10.5.1 Objective:	21
10.5.2 Activities:	21
10.5.3 Deliverables:	22
11 References	22

1 Company Details

1.1 Company Details:

- **Name:** Brain Station 23
- **Website:** <https://brainstation-23.com/> [1]
- **Location:** Plot-02, 8th Floor Bir Uttam Ak Khandaker Rd, Mohakhali C/A, Dhaka, Bangladesh
- **Established:** Founded in 2006
- **Founder:** Raisul Kabir
- **Industry:** Information Technology and Services
- **Specialization:** Provides a wide range of software development and IT solutions
- **Countries Served:** 25+
- **IT Professionals:** 700+
- **Projects Completed:** 2000+
- **Tech Partners:** 12+

1.2 Services Offered:

- Custom Software Development
- Web Development
- Mobile App Development (iOS, Android)
- E-commerce Solutions
- Enterprise Resource Planning (ERP) Solutions
- Cloud Services
- Quality Assurance and Testing
- UI/UX Design Services
- IT Consulting
- Digital Transformation Solutions
- AR & VR Solution
- ML & AI

1.3 Notable Global Partners:

Amazon, Microsoft, Google, Oracle

Technology Partners



Figure 1: Technology Partners

Industries We Serve

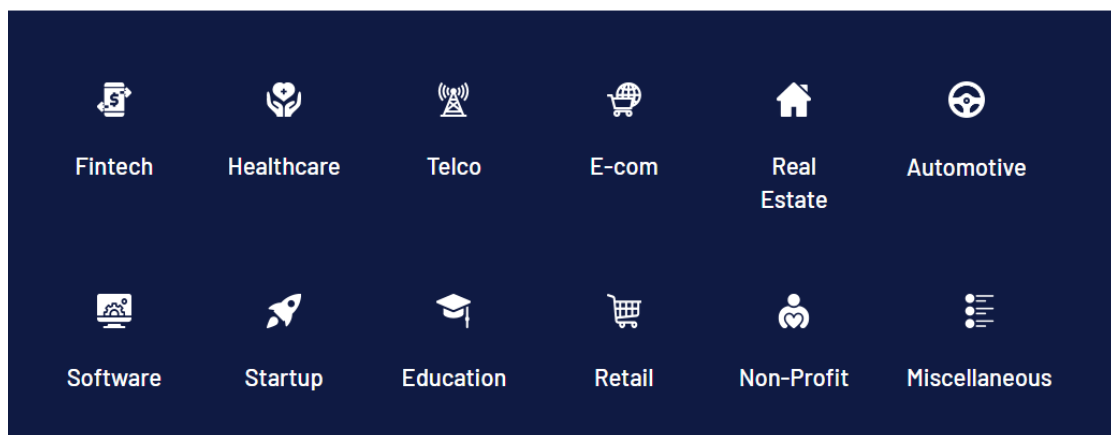


Figure 2: Industries they Serve

Key Client Portfolio

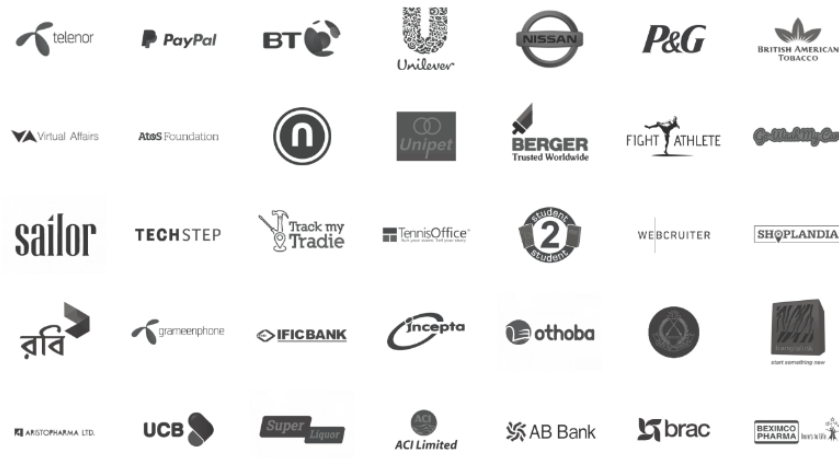


Figure 3: Key Client Portfolio

2 Specific Product / Software Selection

2.1 Introduction:

In the pursuit of adapting to evolving business dynamics and ensuring continuity in trade activities amidst global challenges, the selection of appropriate virtual expo software is pivotal. This chapter delves into the selection of the 3D WebGL-Based Virtual Expo Software, catering specifically to the requirements of CEMS-Global within the trade industry.

2.2 Software Overview:

- **Name:** 3D WebGL-Based Virtual Expo Software
- **Client:** CEMS-Global
- **Platform:** Browser HTML5
- **Industry:** Trade
- **Technology:** Unity, C#, JavaScript, Blender (open-source 3D computer graphics software tool)

2.3 Live Preview

Visit the Live Preview: <https://studio-23.xyz/cems-global-expo-metaverse/>

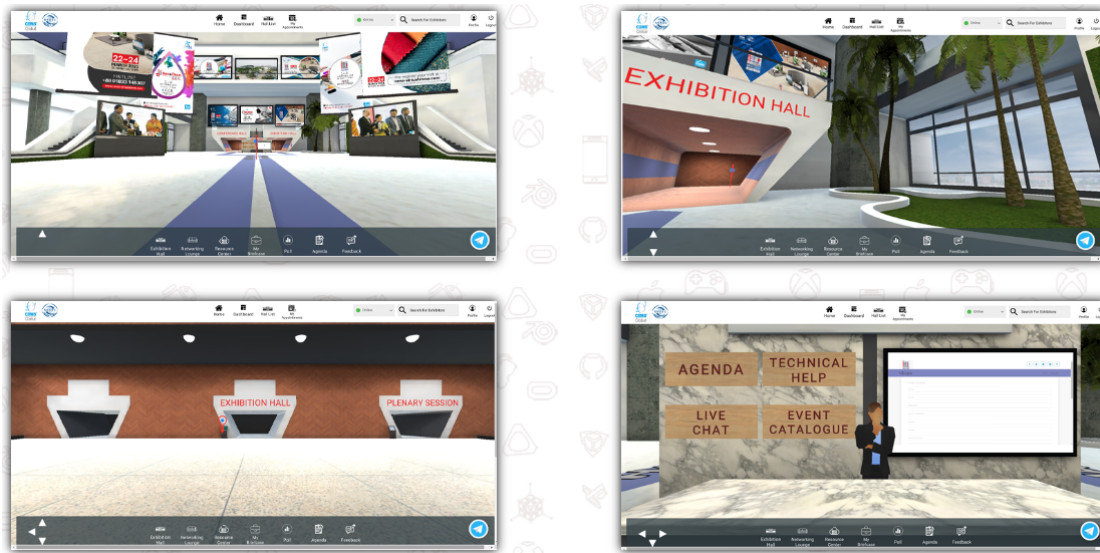


Figure 4:

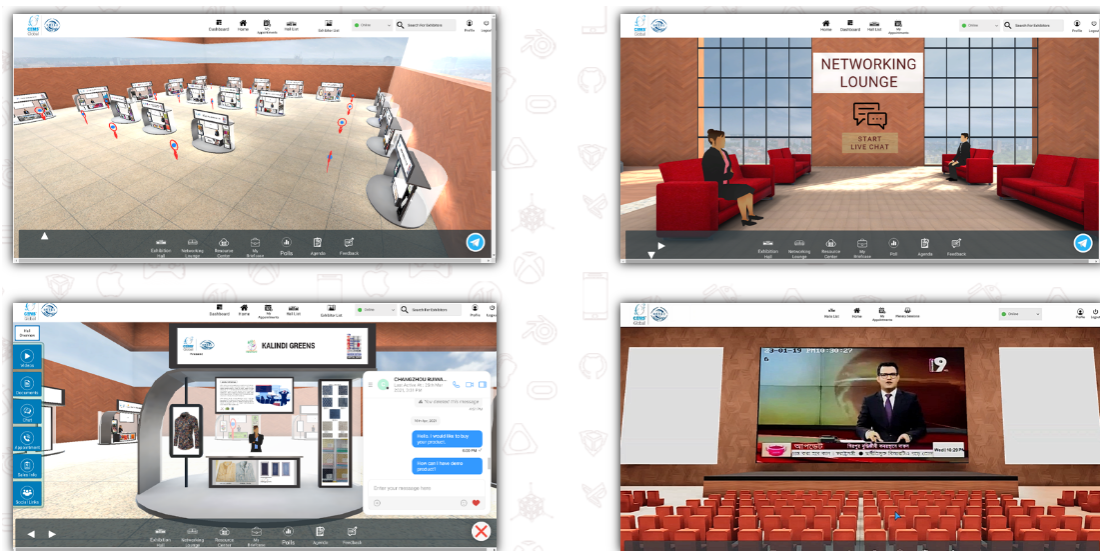


Figure 5:

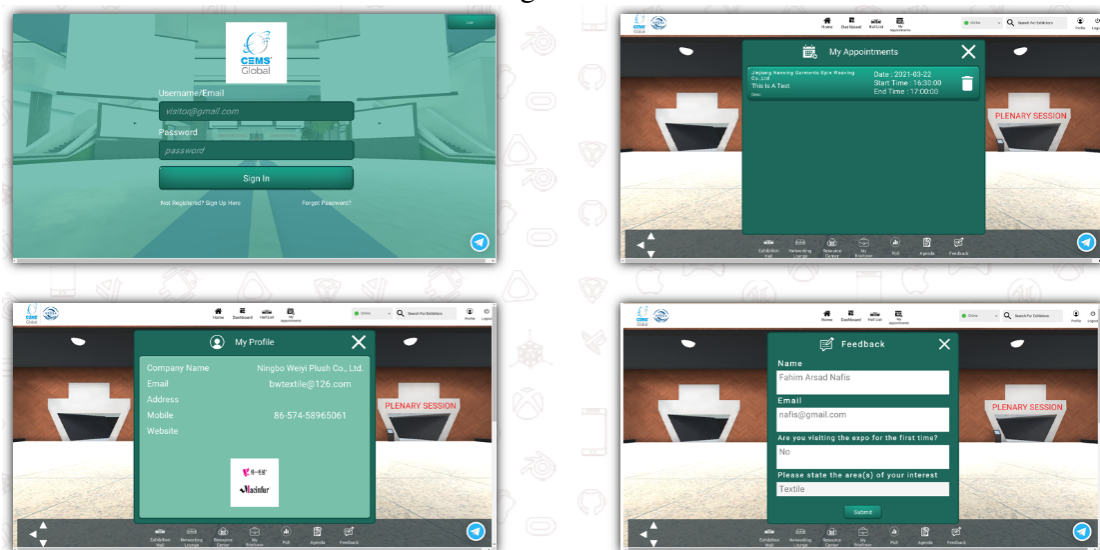


Figure 6:

2.4 Solution:

Brain Station 23 introduces the world's first 3D WebGL-Based Virtual Expo software, addressing the imperative need to bridge the gap between exhibitors and visitors. In light of global preferences for safe and convenient networking avenues, virtual exhibitions emerge as a paramount solution. This service facilitates a seamlessly accessible fair experience with real-world immersion. Key features include instant chatting, audio and video calls, digital brochures, meeting scheduling, and more. Notably, all expo data is configurable through a Cloud Server Backend, ensuring effective admin maintenance and enhanced user experience.

2.5 Description:

A 3D WebGL-Based Virtual Expo Software enables users to host and attend virtual expos or trade shows within a three-dimensional environment using WebGL technology. Leveraging WebGL, a JavaScript API for rendering interactive 3D and 2D graphics, users can engage without the need for additional plugins. Exhibitors can showcase products/services via 3D models, videos, images, and multimedia elements, while attendees navigate through virtual environments, explore booths, and participate in various activities.

2.6 Impact:

Virtual trade fairs facilitate seamless communication between exhibitors and visitors, ushering in a new era of business engagement amidst evolving norms. Exhibitors benefit from comprehensive metrics including attendee trends, webinar engagement, booth visits, and content downloads.

2.7 Features:

- Fully customized 3D architectures for exhibition
- Cloud server backend for data configuration
- Personalized accounts for users
- Easy navigation within the 3D environment
- Corporate networking via instant messaging and calls
- Document collection, contact with representatives, meeting scheduling, and more
- Admin reports providing insights into attendee behavior
- Accessibility to event information, multiple exhibition halls, and virtual seminar facilities
- Appointment management capabilities

2.8 Limitations:

1. **Technology Dependency and Compatibility Issues:** The software's reliance on advanced technologies such as WebGL, Unity 3D, C#, and JavaScript may pose compatibility challenges for users with outdated devices or incompatible browsers, potentially limiting accessibility.
2. **Learning Curve:** Users, particularly those unfamiliar with 3D environments or virtual platforms, may encounter a steep learning curve, leading to decreased engagement, especially among older or less tech-savvy demographics.
3. **Security Concerns:** Hosting the virtual expo on a cloud server backend raises security considerations regarding the storage and handling of sensitive business data. Without robust security measures, there's a risk of data breaches or unauthorized access.
4. **Potential for Technical Glitches:** Despite efforts to ensure smooth navigation, virtual expos remain susceptible to technical glitches and software bugs, which could disrupt the user experience and impact the event's success.

3 Company Employee Details

3.1 Executive Leadership:

- **CEO (Chief Executive Officer):** Responsible for overall strategic direction and decision-making.
- **CTO (Chief Technology Officer):** Oversees the technology strategy and development.
- **CFO (Chief Financial Officer):** Manages financial planning, reporting, and analysis.
- **COO (Chief Operating Officer):** Handles day-to-day operations and ensures efficiency.
- **CIO (Chief Information Officer):** Manages information technology and systems.

3.2 Engineering and Development:

- **Software Engineer:** Designs, develops, and maintains software applications.
- **Frontend Developer:** Focuses on developing the user interface of software applications.
- **Backend Developer:** Works on server-side logic and database management.
- **Full Stack Developer:** Proficient in both frontend and backend development.

- **DevOps Engineer:** Automates and streamlines development processes.
- **Systems Engineer:** Designs and manages complex systems infrastructure.
- **Database Administrator (DBA):** Manages databases, ensures data security, and optimizes performance.
- **Quality Assurance (QA) Engineer:** Tests software to ensure it meets quality standards.
- **UI/UX Designer:** Designs user interfaces and enhances user experience.
- **Technical Lead:** Guides and supervises the technical team.
- **Architect:** Designs the overall structure of software systems.

3.3 Product Management:

- **Product Manager:** Oversees the development and launch of software products.
- **Product Owner:** Represents the customer's interests and priorities in product development.
- **Business Analyst:** Analyzes business processes and requirements.
- **Scrum Master:** Facilitates agile development processes and removes obstacles.

3.4 Design:

- **Graphic Designer:** Creates visual elements for software interfaces.
- **Interaction Designer:** Designs the interactions between users and software systems.

3.5 Data and Analytics:

- **Data Scientist:** Analyzes and interprets complex data sets.
- **Data Analyst:** Extracts insights from data to inform decision-making.
- **Business Intelligence (BI) Analyst:** Analyzes business data to provide insights and support strategic decisions.

3.6 Support and Operations:

- **Technical Support Engineer:** Assists users with technical issues and troubleshooting.
- **Systems Administrator:** Manages and maintains IT infrastructure.
- **Network Engineer:** Designs and maintains network infrastructure.
- **IT Manager:** Oversees IT operations and staff.

3.7 Sales and Marketing:

- **Sales Representative:** Sells software products or services to customers.
- **Sales Manager:** Manages sales teams and strategies.
- **Marketing Manager:** Develops and executes marketing campaigns.
- **Digital Marketing Specialist:** Focuses on digital marketing channels.
- **Content Writer:** Creates written content for marketing purposes.
- **Social Media Manager:** Manages social media presence and engagement.

3.8 Human Resources and Administration:

- **HR Manager:** Manages human resources functions such as recruitment and employee relations.
- **Recruiter:** Sources and hires talent for the company.
- **Office Manager:** Manages administrative tasks and office operations.
- **Administrative Assistant:** Provides support to executives and teams.

3.9 Finance and Accounting:

- **Accountant:** Manages financial records and reports.
- **Financial Analyst:** Analyzes financial data and provides insights.
- **Controller:** Oversees financial reporting and compliance.

3.10 Project Management:

- **Project Manager:** Plans and manages software development projects.
- **Agile Coach:** Provides guidance on agile methodologies and practices.
- **Program Manager:** Oversees multiple projects and their coordination.

3.11 Research and Development:

- **Research Scientist:** Conducts scientific research related to software development.
- **Innovation Manager:** Promotes and facilitates innovation within the company.

3.12 Training and Development:

- **Training Specialist:** Develops and delivers training programs for employees.
- **Learning and Development Manager:** Manages employee learning and development initiatives.

3.13 Content Creation:

- **Technical Writer:** Creates documentation and instructional materials for software products.
- **Documentation Specialist:** Manages documentation processes and standards.

4 Selected Product Model SDLC

4.1 Why Agile SDLC is Used:

The Agile Software Development Life Cycle (SDLC) was chosen for developing the 3D WebGL-Based Virtual Expo Software by Brain Station 23 for several compelling reasons. Agile promotes iterative development, enabling the team to break the project into manageable tasks and deliver functional software in short cycles. This method allows for continuous feedback and adjustments, ensuring the product meets evolving user needs and market demands. By incorporating continuous integration and test-driven development, Agile ensures high-quality code and swift detection of issues. This flexibility and emphasis on collaboration make Agile ideal for a complex, dynamic project like a 3D virtual expo, where rapid adaptation to feedback is crucial.

4.2 12 Agile principles:

The four values of Agile are the pillars of Agile methodology. From those values, the team developed 12 principles. If the four values of Agile are the weight-bearing pillars of a house, then these 12 principles are the rooms you can build within that house. These principles can be easily adapted to fit the needs of your team. The 12 principles used in Agile methodology are:

1. **Satisfy customers through early, continuous improvement and delivery.** When customers receive new updates regularly, they're more likely to see the changes

they want within the product. This leads to happier, more satisfied customers—and more recurring revenue.

2. **Welcome changing requirements, even late in the project.** The Agile framework is all about adaptability. In iterative processes like Agile, being inflexible causes more harm than good.
3. **Deliver value frequently.** Similar to principle 1, delivering value to your customers or stakeholders frequently makes it less likely for them to churn.
4. **Break the silos of your projects.** Collaboration is key in the Agile framework. The goal is for people to break out of their own individual projects and collaborate together more frequently.
5. **Build projects around motivated individuals.** Agile works best when teams are committed and actively working to achieve a goal.
6. **The most effective way to communicate is face-to-face.** If you're working on a distributed team, spend time communicating in ways that involve face-to-face communication like Zoom calls.
7. **Working software is the primary measure of progress.** The most important thing that teams should strive for with the Agile framework is the product. The goal here is to prioritize functional software over everything else.
8. **Maintain a sustainable working pace.** Some aspects of Agile can be fast-paced, but it shouldn't be so fast that team members burn out. The goal is to maintain sustainability throughout the project.
9. **Continuous excellence enhances agility.** If the team develops excellent code in one sprint, they can continue to build off of it the next. Continually creating great work allows teams to move faster in the future.
10. **Simplicity is essential.** Sometimes the simplest solution is the best solution. Agile aims to not overcomplicate things and find simple answers to complex problems.
11. **Self-organizing teams generate the most value.** Similar to principle #5, proactive teams become valuable assets to the company as they strive to deliver value.
12. **Regularly reflect and adjust your way of work to boost effectiveness.** Retrospective meetings are a common Agile practice. It's a dedicated time for teams to look back and reflect on their performance and adapt their behaviors for the future.

5 Total Phase Completion

5.1 5 Phases

These five phases represent the typical stages a software development project goes through, often referred to as the Software Development Life Cycle (SDLC). Here's a breakdown of what happens in each one:

1. **Planning & Analysis:** This is the foundation. You identify the problem the software will solve, research its feasibility, gather requirements from stakeholders (users, clients), and define the project scope and timeline.
2. **Design:** With a clear understanding of the requirements, the software's architecture and user interface (UI) are designed. This could involve creating wireframes, flowcharts, and other visual representations to illustrate how the software will function.
3. **Implementation (or Development):** This is where the coding happens! Developers translate the designs into actual code, building the core functionalities of the software. Here, following best practices and code quality standards is crucial.
4. **Testing:** Once the software is built, it's rigorously tested to identify and fix bugs (errors) and ensure it meets the defined requirements. Different testing approaches may be used, like functional testing (does it do what it's supposed to?) and usability testing (is it user-friendly?).
5. **Maintenance:** Software doesn't exist in a vacuum. New issues may arise, or there might be a need for updates and improvements based on user feedback. The maintenance phase involves fixing bugs, adding new features, and ensuring the software continues to function as intended over time.

5.2 Gantt Chart

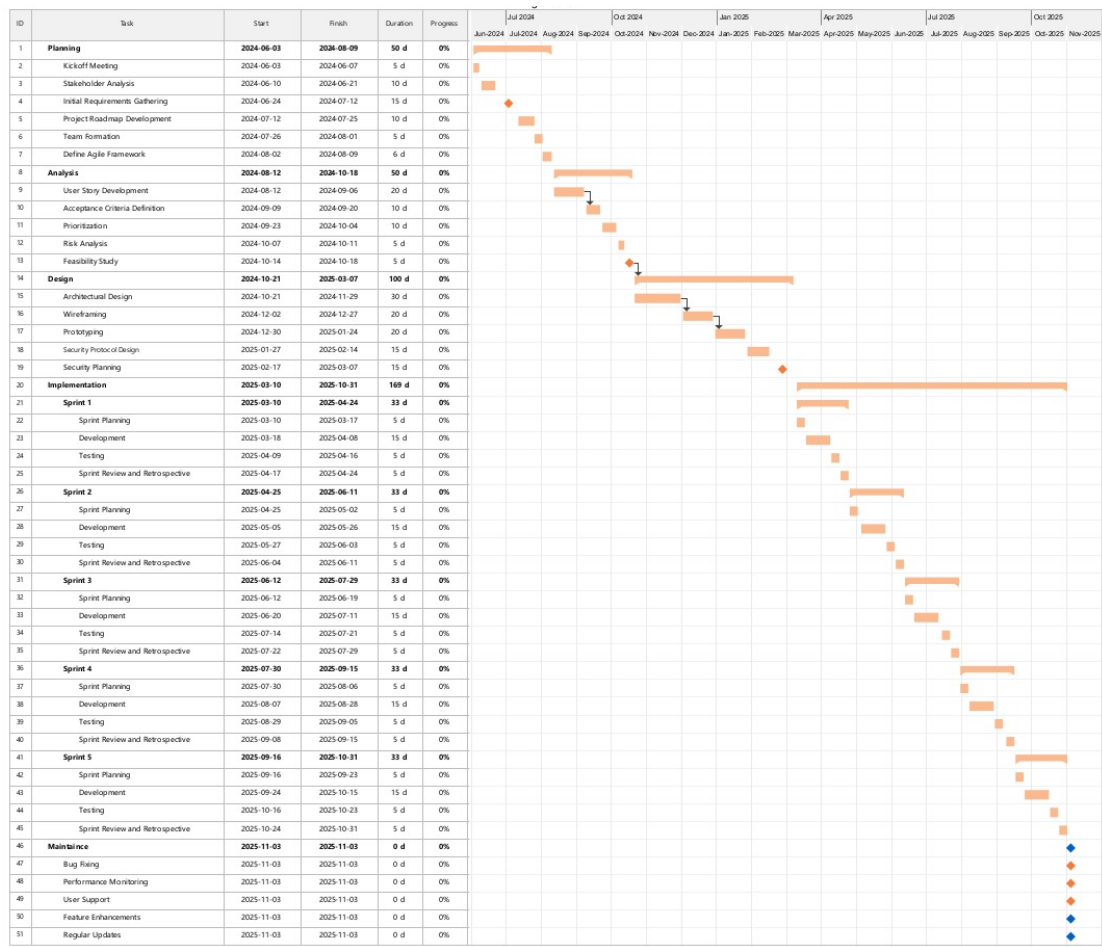


Figure 7: Gantt Chart

6 Planning

6.1 Objectives:

- Define the project’s scope, goals, and objectives.
- Identify key stakeholders and their roles.
- Establish a high-level timeline and budget.
- Gather initial requirements and prioritize them.

6.2 Activities:

1. **Kickoff Meeting:** Gather all stakeholders to discuss the project’s vision and goals.
2. **Stakeholder Analysis:** Identify all stakeholders and their requirements.

3. **Initial Requirements Gathering:** Conduct interviews, surveys, and workshops to gather preliminary requirements.
4. **Project Roadmap:** Develop a high-level project roadmap outlining major milestones and deliverables.
5. **Team Formation:** Assemble the project team, defining roles and responsibilities.
6. **Define Agile Framework:** Decide on the specific Agile methodology (e.g., Scrum, Kanban) and tools (e.g., JIRA, Trello) to be used.

6.3 Deliverables:

- Project Charter
- High-level project plan
- Initial product backlog
- Stakeholder register
- Agile process guidelines

7 Analysis

7.1 Objectives:

- Refine requirements and create detailed user stories.
- Develop acceptance criteria for each user story.
- Identify potential risks and develop mitigation strategies.

7.2 Activities:

1. User Story Development: Work with stakeholders to create detailed user stories.
2. Acceptance Criteria Definition: Define clear, testable acceptance criteria for each user story.
3. Prioritization: Prioritize user stories based on business value and technical feasibility.
4. Risk Analysis: Identify potential project risks and create a risk management plan.
5. Technical Feasibility Study: Conduct a feasibility study to ensure the project is technically achievable.

7.3 Deliverables:

- Detailed user stories and acceptance criteria
- Prioritized product backlog
- Risk management plan
- Feasibility study report

8 Design

8.1 Objectives:

- Create a detailed design for the system architecture and user interface.
- Develop wireframes and prototypes for key components.
- Plan for data storage and security.

8.2 Activities:

1. **Architectural Design:** Design the system architecture, including the server, client, and database components.
2. **Wireframing:** Create wireframes for the user interface, focusing on user experience and navigation.
3. **Prototyping:** Develop interactive prototypes to validate design choices with stakeholders.
4. **Database Design:** Design the database schema to support the application's data needs.
5. **Security Planning:** Plan for data security, including encryption, authentication, and authorization mechanisms.

8.3 Deliverables:

- System architecture diagrams
- Wireframes and UI/UX designs
- Interactive prototypes
- Database schema
- Security plan

9 Implementation

9.1 Implementation Phase:

During the implementation phase of the 3D WebGL-Based Virtual Expo Software, Brain Station 23 leveraged Agile principles to ensure an efficient and effective development process. The team worked in short, iterative cycles, typically lasting 1-2 weeks, allowing them to deliver incremental updates and continuously refine the software. Each iteration involved planning, coding, testing, and reviewing, ensuring that the software evolved smoothly and steadily.

Key activities during this phase included continuous integration, where new code was frequently integrated into the existing codebase. This practice minimized integration issues and allowed for immediate testing and quality assurance. Automated build and deployment tools facilitated this process, ensuring that any new feature or bug fix was promptly tested and deployed.

Test-driven development (TDD) was another critical practice. Developers wrote automated tests before coding each new feature, ensuring that all functionality was thoroughly tested and met quality standards. This approach reduced the likelihood of bugs and streamlined the development process by catching issues early.

Throughout the implementation, the team maintained close communication and collaboration with stakeholders. Regular meetings and updates ensured that everyone was aligned on project goals and priorities, allowing the team to quickly address any challenges or changes in requirements. This collaborative and iterative approach enabled Brain Station 23 to deliver a high-quality, user-centered product that effectively bridged the gap between exhibitors and visitors in a virtual expo environment. By focusing on continuous feedback and adaptability, the Agile SDLC model allowed for a successful, on-time delivery of the 3D WebGL-Based Virtual Expo Software.

9.2 Objectives:

- Develop the software in iterative sprints.
- Conduct regular testing to ensure quality and functionality.
- Continuously integrate and deploy new features.

9.3 Activities:

- **Sprint Planning:** Plan each sprint, selecting user stories from the backlog to be developed.
- **Development:** Implement user stories, following coding standards and best practices.
- **Testing:** Conduct unit tests, integration tests, and user acceptance tests for each sprint.

- **Continuous Integration/Continuous Deployment (CI/CD):** Set up a CI/CD pipeline to automate the build, testing, and deployment process.
- **Sprint Review and Retrospective:** Review completed work with stakeholders and conduct retrospectives to improve the process.

9.4 Deliverables:

- Incremental builds of the software
- Test reports
- CI/CD pipeline setup
- Sprint review notes
- Retrospective notes

10 Maintenance

10.1 Four major activities:

The Process of Maintaining Information Systems.

Maintain and enhance the information system to ensure it continues to meet the users' needs and operates efficiently. There are four major activities:

1. Obtaining maintenance requests
2. Transforming requests into changes
3. Designing changes
4. Implementing changes

10.2 Obtaining maintenance requests

10.2.1 Objective:

- Gather and prioritize requests for system maintenance from users and stakeholders.

10.2.2 Activities:

1. **Request Submission:** Provide a formal mechanism (e.g., help desk, ticketing system) for users to submit maintenance requests.

2. **Request Logging:** Record each request with detailed information, including the requester's contact information, the nature of the problem or enhancement, and the urgency of the request.
3. **Initial Assessment:** Conduct an initial assessment to determine the validity and priority of each request. Categorize requests into corrective, adaptive, perfective, or preventive maintenance.
4. **Stakeholder Review:** Review the requests with relevant stakeholders to ensure they align with business needs and priorities.

10.2.3 Deliverables:

- Maintenance request log
- Categorized and prioritized list of requests
- Initial assessment report

10.3 Transforming requests into changes

10.3.1 Objective:

- Analyze and convert maintenance requests into actionable change requirements.

10.3.2 Activities:

1. **Requirements Analysis:** Conduct a detailed analysis of the maintenance request to understand the underlying issues and the desired changes.
2. **Feasibility Study:** Evaluate the feasibility of the requested changes, considering technical, operational, and financial aspects.
3. **Change Proposal:** Develop a change proposal that includes a detailed description of the required changes, potential impacts, and estimated effort.
4. **Approval Process:** Present the change proposal to the change control board or relevant decision-makers for approval.

10.3.3 Deliverables:

- Detailed requirements document
- Feasibility study report
- Change proposal
- Approved change requests

10.4 Designing changes

10.4.1 Objective:

- Design the necessary changes to the system based on the approved maintenance requests.

10.4.2 Activities:

1. **System Design:** Create detailed design specifications for the changes, including updates to data models, business logic, user interfaces, and integration points.
2. **Design Review:** Conduct design review sessions with stakeholders and team members to ensure the proposed changes meet requirements and adhere to design standards.
3. **Prototyping:** Develop prototypes or mockups for complex changes to validate the design approach with users and stakeholders.
4. **Documentation:** Update all relevant design and system documentation to reflect the planned changes.

10.4.3 Deliverables:

- Detailed design specifications
- Design review notes and feedback
- Prototypes/mockups
- Updated design documentation

10.5 Implementing changes

10.5.1 Objective:

- Execute the designed changes in the information system and ensure they function correctly.

10.5.2 Activities:

1. **Development:** Code the changes according to the design specifications. This includes updates to software, databases, and configuration settings.
2. **Testing:** Perform thorough testing, including unit testing, integration testing, and user acceptance testing, to ensure the changes work as intended without introducing new issues.

3. **Deployment:** Deploy the changes to the production environment following established deployment procedures, ensuring minimal disruption to users.
4. **Documentation Update:** Update user manuals, help files, and other relevant documentation to reflect the changes.
5. **Training:** Provide training to users, if necessary, to ensure they understand how to use the new or updated features.
6. **Post-Implementation Review:** Conduct a post-implementation review to assess the effectiveness of the changes and identify any further improvements.

10.5.3 Deliverables:

- Updated codebase and configuration
- Test plans and test results
- Deployment plan and records
- Updated user documentation
- Training materials
- Post-implementation review report

By following these structured activities, the process of maintaining information systems ensures that the system remains functional, efficient, and aligned with user needs and organizational goals.

11 References

References

- [1] Brain Station 23. <https://brainstation-23.com/>. [Last updated on Jan 17, 2024].