**Functional Design Document: Amazebot – Personalized Learning Journeys for New Employees**

**Version 1.0**

**Date:** June 17, 2025  
**Prepared by:** Srikar & Sudeshna  
**Approved by:** [Dev Team]

**Table of Contents**

1. Project Overview
2. User Roles & Profiles
3. Learning Journey Stages
4. Amazebot Functional Feature Set
5. System Architecture
6. Data Model & Storage
7. Non-Functional Requirements
8. User Interface (UI) & User Experience (UX)
9. Technical Specification + API Design Document
10. Workflows & System Behaviour
11. Security & Compliance
12. Reporting & Analytics
13. Open Questions & Next Steps
14. System-Level Capabilities
15. Azure & MCP Integration Mapping
16. Integration with Sharepoint
17. Deployment and Maintenance
18. Appendices

**1. Project Overview**

**1.1 Project Name:** Amazebot – Personalized Learning Journey for New Employees  
**1.2 Client:** Amazech Employees  
**1.3 Developed by:** Amazech Technologies  
**1.4 Objective:** To design an AI-driven chatbot and LLM-based employee training platform that guides new hires through a structured learning path in respective applications.

**1.5 Purpose:** This document outlines the functional design of an internal training ecosystem aimed at enhancing employee onboarding, client-specific training, and organization-wide learning initiatives. The system leverages Model Context Protocol (MCP) and Azure services to provide a personalized, scalable, and intelligent learning environment.

**1.6 Scope**

* **In-Scope**: Internal use within Amazech; includes onboarding, training, and learning initiatives.
* **Out-of-Scope**: Client-facing modules and external training platforms.
* **Key Components**:
  + Centralized learning assistant: AmazeBot
  + Integration with Microsoft tools: Azure SQL, Blob Storage, SharePoint, Outlook, Teams
  + Real-time analytics and AI-driven mentoring
  + Document-driven automation using MCP context stores

**2. User Roles & Profiles**

| **Role** | **Description** | **Key Permissions** |
| --- | --- | --- |
| **Admin** | Manages users, training assignments, and content | Assign training, track completions, override statuses |
| **Trainer** | Develops and reviews training modules | Upload content, initiate review workflows |
| **SME (Subject Matter Expert)** | Provides expert validation for training content | Review and approve training modules |
| **Legal** | Ensures compliance of training materials | Approve final training content |
| **Employee** | Engages with assigned training modules | Access modules, complete quizzes, track progress |

**3. Learning Journey Stages**

**3.1 Onboarding & Orientation**

* **Objective:** Acquaint new employees with Amazech’s and culture, policies, and domain.
* **Key Components:**
  + Introduction to Amazech’s mission, values, and organizational structure.
  + Overview of Amazech’s offerings.
  + Compliance, regulatory standards, and ethical guidelines.
  + Introduction to domain.

**3.2 Foundational Training**

* **Objective:** Equip employees with basic knowledge and skills required for their roles.
* **Key Components:**
  + Understanding domain related terminology and concepts.
  + Basic functionalities of the services offered on the application.
  + Basic data entry and processing tasks.

**3.3 Intermediate Training**

* **Objective:** Enhance proficiency and introduce more complex tasks.
* **Key Components:**
  + Advanced features of the respective application.
  + Business process flow of the application on which the employees are getting trained.
  + Setup-related knowledge like Product, Program, custom fields, custom flows etc.
  + Complete process flow from client onboarding to termination.

**3.4 Advanced Training**

* **Objective:** Develop expertise and leadership capabilities for decision-making.
* **Key Components:**
  + Strategic decision-making in the respective application.
  + Leadership and mentoring skills.
  + Advanced analytics and reporting based on the application functions.
  + Project management and cross-functional collaboration.

**4. Amazebot Functional Feature Set**

**4.1 Personalized Learning Journeys**

* **AI-Driven Assessment:** Amazebot assesses individual learning styles and progress.
* **Adaptive Content Delivery:** Customizes training content based on assessment results.

**4.2 Micro-Learning Modules**

* **Bite-Sized Content:** Delivers content in manageable chunks to enhance retention.
* **Interactive Elements:** Includes quizzes and interactive scenarios to reinforce learning.

**4.3 Interactive Simulations**

* **Real-World Scenarios:** Incorporates case studies and simulations for practical experience.
* **Risk-Free Environment:** Allows employees to practice decision-making without consequences.

**4.4 Continuous Feedback Mechanisms**

* **Instant Feedback:** Provides immediate feedback on tasks and quizzes.
* **Progress Tracking:** Monitors learner progress and identifies areas for improvement.

**4.5 Integration with Amazech Sharepoint**

* **Seamless Integration:** Ensures tracking of progress and performance through integration with Amazech Sharepoint.
* **Data Synchronization:** Synchronizes user data, course completion, and assessment results with the training details loaded in sharepoint.

**5. System Architecture**

* **Frontend:** Web-based interface accessible via browsers.
* **Backend:** Cloud-based server infrastructure with scalable resources.
* **Database:** Relational database for storing user data, course materials, and progress tracking.
* **AI Engine:** Integration with OpenAI's ChatGPT for natural language processing and personalized interactions.
* **Sharepoint Integration:** Integration with Amazech's sharepoint source for data synchronization.

**5.1 User Management**

* **Admin Capabilities**:
  + Create users manually or via bulk CSV import.
  + Assign roles and training modules.
  + Override training completion statuses.
  + View and manage user progress.
* **Data Model**:

json

CopyEdit

{ "user": { "id": "E001", "role": "Finance Analyst", "client": "ABC Corp" }, "training\_status": { "completed": ["mod-fin-001"], "in\_progress": ["mod-fin-002"] } }

**5.2 Training Module Management**

* **Module Lifecycle**:
  + Upload training materials (PDFs, slides, SCORM, videos).
  + Auto-summarize content using LLMs (with human review).
  + Implement version control: draft, published, retired.
  + Track review workflows: Trainer → SME → Legal.
* **MCP Context Example**:

json

CopyEdit

{ "module": { "id": "mod-fin-001", "client": "ABC Corp", "status": "published", "last\_reviewed\_by": "Legal" } }

**5.3 AmazeBot AI Integration**

* **Capabilities**:
  + Provide real-time FAQ responses.
  + Explain quiz answers and concepts.
  + Suggest personalized learning paths.
  + Offer feedback on training progress.
* **MCP Context Example**:

json

CopyEdit

{ "session": { "user\_id": "E001", "current\_module": "mod-fin-002", "lang": "en-US" } }

**5.4 Notification System**

* **Triggers**:
  + Email notifications upon training assignment.
  + Outlook calendar invites for training sessions.
  + Weekly digest emails for users behind schedule.
  + Reminders based on MCP timestamp checks.

**6. Data Model & Storage**

* **Azure SQL**: Stores user metadata, module statuses, and role mappings.
* **Blob Storage**: Holds training materials.
* **Azure Table/NoSQL**: Manages MCP context data.
  + Example:

json

CopyEdit

{ "user\_id": "E001", "context": { "role": "Finance Analyst", "module\_progress": ["mod-fin-001"] } }

**7. Non-Functional Requirements**

* **Performance**:
  + 99.9% system uptime.
  + AI response time under 1 second.
* **Security**:
  + Data encryption at rest and in transit.
  + Role-based access control.
  + Single Sign-On (SSO) via Azure AD.
* **Scalability**:
  + Azure regional failover and disaster recovery capabilities.

**8. User Interface (UI) & User Experience (UX)**

* **Dashboard:** Personalized dashboard displaying learning path, progress, and upcoming modules.
* **Chat Interface:** Conversational interface powered by ChatGPT for interactive learning.
* **Notifications:** Alerts for upcoming modules, assessments, and feedback.
* **Accessibility:** Ensure compliance with accessibility standards for users with disabilities.
* **Admin Panel**: Interface for managing users, training assignments, and content.
* **Quiz Interface**: Engaging UI for quiz participation with instant feedback.

*Note: Detailed wireframes and UI designs will be developed in Figma.*

**9. Technical Specification + API Design Document**

**System Overview**

* Frontend: React (with TailwindCSS)
* Backend: Node.js (Express)
* Database: Azure SQL + Azure Table or Pinecone for MCP
* LLM Integration: Simulated ChatGPT using local logic, ready to swap with OpenAI API
* Auth: Azure AD (mocked in starter)

**9.1 Modules & APIs**

* **User Management**

|  |  |  |  |
| --- | --- | --- | --- |
| **Feature** | **Method** | **Endpoint** | **Description** |
|  |  |  |  |
| Register User | POST | /api/users | Create a new user |
| Get User | GET | /api/users/:id | Retrieve user data |
| Assign Training | POST | /api/users/:id/assign | Assign module(s) to a user |
| Get Progress | GET | /api/users/:id/progress | Fetch current training status |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

* **Training Module Management**

|  |  |  |  |
| --- | --- | --- | --- |
| **Feature** | **Method** | **Endpoint** | **Description** |
| Upload Module | POST | /api/modules | Upload training content (PDF/SCORM etc.) |
| Get Modules | GET | /api/modules | List available modules |
| Review Workflow | POST | /api/modules/:id/review | Submit for SME/Legal review |
| Publish Module | POST | /api/modules/:id/publish | Finalize and publish content |

* **Quiz Engine**

|  |  |  |  |
| --- | --- | --- | --- |
| **Feature** | **Method** | **Endpoint** | **Description** |
| Submit Quiz | POST | /api/quiz/:moduleId | Submit answers |
| Get Feedback | GET | /api/quiz/:moduleId/feedback | Simulated feedback from "Amazebot" |

* **MCP Context API**

|  |  |  |  |
| --- | --- | --- | --- |
| **Feature** | **Method** | **Endpoint** | **Description** |
| Get Context | GET | /api/context/:userId | Retrieve user+module context bundle |
| Update Context | POST | /api/context/:userId | Store progress, feedback, quiz stats |
| Trigger LLM | POST | /api/amazebot | Simulated or real ChatGPT call using MCP bundle |

* **Notifications**

|  |  |  |  |
| --- | --- | --- | --- |
| **Feature** | **Method** | **Endpoint** | **Description** |
| Send Email | POST | /api/notifications/email | Simulated email reminder |
| Sync Outlook | POST | /api/integrations/outlook | Placeholder for Outlook calendar invite |

**10. Workflows & System Behaviour**

**10.1 Training Assignment Workflow**

1. **Admin** assigns training to a user.
2. **MCP** logs context: user, module, and assignment timestamp.
3. **System** sends email notification and Outlook calendar invite.
4. **AmazeBot** provides reminders and progress tracking.

**10.2 Module Review Workflow**

1. **Trainer** uploads training content.
2. **MCP** tags content as "draft".
3. **SME** reviews and provides feedback.
4. **Legal** reviews for compliance.
5. **MCP** updates status to "published" upon final approval.

**10.3 Quiz Interaction Workflow**

1. **User** completes quiz associated with a module.
2. **MCP** records response times, errors, and confidence levels.
3. **AmazeBot** provides instant feedback and suggests areas for improvement.

**11. Security & Compliance**

* **Authentication**: Azure AD SSO for secure access.
* **Authorization**: Role-based access control to ensure appropriate permissions.
* **Audit Trails**:
  + Exportable logs for user activities.
  + Visualized dashboards for compliance monitoring.
* **Content Review**:
  + Documented review trails: SME → Legal.
  + MCP sanitization layer to ensure content integrity.

**12. Reporting & Analytics**

* **Product Owner Dashboard**:
  + Training completion rates.
  + Module difficulty assessments.
  + Quiz performance analytics.
  + AI token usage statistics.
  + Identification of performance outliers.

**13. Open Questions & Next Steps**

* **Branding**: Will training modules require client-specific branding?
* **Gamification**: Should quiz feedback include gamified elements (e.g., streaks)?
* **MCP Tuning**: Is there a need for department-specific MCP context customization?

**14. System-Level Capabilities**

* **Customization**:
  + Adjustable pass thresholds for quizzes.
  + Configurable learning paths based on user roles.
  + Set limits on quiz retakes.
  + Visual training heatmap calendar.
  + AI-driven training suggestions based on performance data.
  + Document lifecycle tracking with MCP tagging.

**15. Azure & MCP Integration Mapping**

| **Azure Service** | **Role in System** | **MCP Interaction Model** |
| --- | --- | --- |
| **WebApp** | Hosts UI and admin panel | Injects session context bundle for each page view |
| **SQL DB** | Stores user, role, and module data | Feeds structured input to MCP bundles |
| **Blob Storage** | Stores training media files | File references passed via MCP for LLM awareness |
| **Table/NoSQL** | Stores MCP JSON bundles | Used directly by AmazeBot and dashboards |
| **Functions** | Handles automation tasks (e.g., emails) | MCP triggers used for timing/context updates |
| **OpenAI (AmazeBot)** | Provides chatbot and summarization logic | MCP supplied with embedded structured context |
| **SharePoint** | Manages document lifecycle and reviews | MCP links to final approved documents |
| **Azure AD** | Manages identity and SSO | Contextualizes roles within the MCP system |

**16. Integration with Sharepoint**

* **User Data Synchronization:** Sync user profiles, roles, and permissions between Amazebot and Sharepoint.
* **Course Enrolment:** Automate enrolment of users into appropriate training modules based on their roles.
* **Progress Tracking:** Update course completion status and assessment results in real-time.
* **Reporting:** Generate reports on learner progress and performance for administrators.

**17. Deployment and Maintenance**

* **Deployment Strategy:** Use a phased deployment approach, starting with a pilot group before full-scale rollout.
* **Training:** Provide training sessions for administrators and end-users.
* **Support:** Offer ongoing technical support and maintenance services.
* **Updates:** Regularly update the system with new features, content, and security patches.

**18. Appendices**

**Appendix A: Glossary of Terms**

* **AI:** Artificial Intelligence
* **LLM:** Large Language Model
* **GPT:** Generative Pre-trained Transformer
* **MCP:** Model Context Protocol