Task 1 Answer

**Part A – Analysis**

1. **Introduction**

The purpose of this document is to analyse the needs of Gibjohn Tutoring and propose a digital solution that addresses their requirements. This includes exploring existing platforms, identifying functional & non-functional requirements, and ensuring legal, regulatory, and professional compliance.

**Client:** GibJohn Tutoring provides face to face tutoring and learning resources.

**Objectives:** Create a digital platform that supports interactive resources, wider learning, & progress monitoring.

1. **Analysis of existing solutions**

Strengths:

**Khan Academy** >> strong variety of content and assessments

**Duolingo** >> effective gamification

***Weaknesses:***

* Limited accessibility options. (Duolingo does not support those with visual impairments 100% effectively)
* Lack of deep personalisation. (Khan & Duolingo - All personalised content is standardised, and custom personalisation is not permitted)

***Opportunities:***

* Introduce personalisation (potentially AI-driven).
* Accessibility-first design.
* Gamified rewards tied to learning progress.

***Justification:*** *A new solution is needed to combine the strengths of existing platforms with accessibility, compliance, and tailored learning for GibJohn’s audience.*

1. **Emerging Technologies**

***AI:*** Adaptive quizzes, automated marking**.**

***IoT*:** Smart classrooms linking to the platform.

***AR/VR:*** Virtual lab experiments and immersive subject exploration.

***Cloud computing:*** Scalable, cost-effective hosting.

***Integration:*** *Cloud included immediately; AI, IoT & AR/VR reserved for later qiterations*

1. **Business Context**

**Overview:** Tutoring company expanding digitally

**Goals:** Wider: access, improved engagement

**Audience:** Learners (school-aged), tutors, guardians.

**Constraints:** Time, compliance with GDPR/Equality Act, phased implementation

1. **Requirements analysis**

**Functional Requirements**

* Learner registration & login
* Tutor uploads and resource sharing
* Progress tracking dashboards
* Reward system for achievements.

**Non-Functional Requirements**

* Secure (password hashing, HTTPS).
* Accessible (WCAG 2.1).
* Reliable (99.9% uptime)
* Usable (simple navigation, responsive)

**KPIs**

* <5s page load time
* 99.9% uptime
* 80% weekly active learner usage
* 90% satisfaction in UAT.

**User Acceptance Criteria (UACs):**

* **UAC-1:** Learner can create an account with valid details.
* **UAC-2:** Learner can log in securely with email and password.
* **UAC-3:** Tutor can upload resources and learners can access them.
* **UAC-4:** Learner dashboard shows progress (%) for each subject.
* **UAC-5:** Progress updates automatically after a quiz is submitted.
* **UAC-6:** Reward badge appears when a learner meets criteria (e.g., score ≥80 three times).
* **UAC-7:** The system works on mobile and desktop, with all functions available via keyboard navigation.
* **UAC-8:** Page load time is under 5 seconds (assessed in staging).

**MoSCoW**

* **Must have:** Login, progress tracking, content delivery.
* **Should have:** Rewards and accessibility features.
* **Could have:** AI-driven personalisation.
* **Won’t have:** VR integration at launch.

1. **Problem Decomposition**

**Modules:**

* Authentication (validated register, validated login, functional sessions, and database)
* Resource Management (static assets, caching, image/media optimisation)
* Progress tracking (lessons correct answers wrong answers, unanswered)
* Reward System (wallpapers, banners)
* Admin Dashboard (max clearance access)

**System Architecture:**

**Frontend (html, CSS) >> Backend (Flask) >> Database>> (SQLite)**

**Hosting:** Heroku

**Security:** Encrypted logins, HTTPS, role-based access.

**Emerging Tech:** AI for adaptive learning, cloud for scalability.

***Justification:*** *Secure, scalable, industry-standard technologies aligned to GibJohn’s needs.*

**System Flow Diagram**

[Start: Home/Login]

|

v

[Register] -----> [Login]

| |

v v

[Learner Dashboard] <---- [Tutor Dashboard]

| |

v v

[View Content/Take Quiz] [Upload Resources/Create Assignment]

| |

v v

[Progress Tracking] [View Learner Progress]

|

v

[Rewards System]

|

v

[Logout]

1. **Legal & Regulatory Compliance**

|  |  |  |
| --- | --- | --- |
| Law/Standard | Requirement | How addressed |
| GDPR | **Protect learner data** | **Encryption, consent, anonymisation** |
| Equality Act 2010 | **Accessibility** | **WCAG2.1 compliance, screen reader support** |
| Consumer Rights Act | **Transparent, fair terms** | **Clear T&Cs, opt-out options** |
| Copyright/IP law | **Respect ownership** | **Licensed resources, attribution** |
| WC3 standards | **Accessible web design** | **Semantic HTML, ARIA roles** |

1. **Professional & Ethical Standards**

**Aligned with BCS Code of Conduct:**

* + **Public interest:** Accessibility-first design.
  + **Integrity:** Transparent handling of learner data.
  + **Competence:** Secure, reliable development practices.
  + **Privacy:** GDPR-compliant data storage.

1. **Justification of Design Choices**

**HTML:** Using **HTML5 semantic landmarks** (<header>, <nav>, <main>, <footer>) ensures the site is screen-reader friendly and meets **WCAG 2.1** accessibility standards. This reduces reliance on JavaScript for structure, improving page load performance and helping achieve the KPI of **<5s load times**.

**CSS:** Using **CSS Grid and Flexbox** provides responsive layouts that adapt across phones, tablets, and desktops. A consistent design system (tokenised colours, rem-based typography) maintains readability and ensures **WCAG contrast ratios** for accessibility.

**JavaScript (tiny):** Only a small amount of JavaScript is used for **form validation** and **small interactions** (such as showing error messages or toggling UI elements). Core functionality still works without JavaScript, which makes the prototype more reliable and accessible on older devices.

**Flask:** Flask with **Blueprints** (auth/resources/progress), **Jinja2 templates**, **Flask-Login** for authentication, **SQLAlchemy ORM**, and **CSRF protection** provides a secure and testable backend. Using Python also keeps the door open for future integration with **AI features** such as adaptive learning.

**Heroku:** Chosen as a simple, cloud-based hosting option because it provides:

* **Automatic HTTPS** → secure logins and data transfer.
* **Easy GitHub deployment** → fast updates and version control.
* **Scalability** → can handle more users if GibJohn grows.
* **Monitoring/logs** → helps track uptime and fix errors quickly.

This makes Heroku a safe, reliable platform that meets the client’s needs without complex setup.

1. **Risk Assessment**

|  |  |  |  |
| --- | --- | --- | --- |
| Risk | Impact | Likelihood | Mitigation |
| Data Breach | **High** | **Medium** | **Encryption, (2FA potentially)** |
| Missed Deadlines | **Medium** | **High** | **Agile sprint, weekly reviews** |
| System Downtime | **High Impact** | **Low** | **Cloud redundancy** |
| Low learner uptake | **Medium** | **Medium** | **Gamification, user testing** |
| Copyright Infringement | **High** | **Medium** | **Used licenced materials only** |