**PROJECT NAME: Smart Home Lightning System**

**GROUP MEMBERS: Celal KURT, Eda Naz SOYTÜRK, Efe Kaan ÇAĞLAYAN, Eren AYAS, Hakan Baha GÜNEŞLİLER, Kayra YEŞİLKAYA**

|  |  |
| --- | --- |
| TASK # | PROJECT TASKS WHICH REQUIRE SOFTWARE TOOL SUPPORT |
| 1 | Object Recognition and Tracking System |
| 2 | Database Integration |
| 3 | User Behavior Tracking and Personalization |
| 4 | Smart Light System Integration |
| 5 | Mobile Application Development |

|  |
| --- |
| SOFTWARE TOOLS FOR TASK 1: **OBJECT RECOGNITION AND TRACKING SYSTEM** |
| Tool Cost/Training/Functionality Data   |  |  |  |  |  | | --- | --- | --- | --- | --- | | Tool | SentiSight.ai | Azure Custom Vision Service | Amazon Rekognition | Google Cloud Vision AI | | Cost | 2725 $ | 6850 $ | 3850 $ | 3748.5 $ | | Training Days | 18 | 21 | 10 | 12 | | Functionality | 85 | 90 | 80 | 87 |   Normalized Cost/Training/Functionality Data   |  |  |  |  |  | | --- | --- | --- | --- | --- | | Tool | SentiSight.ai | Azure Custom Vision Service | Amazon Rekognition | Google Cloud Vision AI | | Cost | 40.37 | 100 | 57.04 | 55.5 | | Training Days | 85.7 | 100 | 47.6 | 57.1 | | Functionality | 94.4 | 100 | 88.8 | 96.6 |   Normalized Tool Graph |
| Which tool has been selected? Why?  For our project, we have selected Azure Custom Vision. It offers strong functionality with a score of 90, making it a reliable tool for image recognition. Although it has the highest cost at 6850 USD, it provides advanced features that are essential for our project. The training time is 21 days, which is longer than some other tools, but still reasonable considering the complexity of the task. Despite its higher cost and training time, Azure Custom Vision is a good choice because of its robust capabilities and the level of customization it offers for our needs. |

|  |
| --- |
| SOFTWARE TOOLS FOR TASK 2: DATABASE INTEGRATION |
| Tool Cost/Training/Functionality Data   |  |  |  |  |  | | --- | --- | --- | --- | --- | | Tool | Azure SQL Database | FireBase | Google Cloud SQL | MongoDB  Atlas | | Cost | 2650 $ | 1500 $ | 2600 $ | 2850 $ | | Training Days | 16 | 15 | 30 | 18 | | Functionality | 90 | 85 | 85 | 95 |   Normalized Cost/Training/Functionality Data   |  |  |  |  |  | | --- | --- | --- | --- | --- | | Tool | Azure SQL Database | Firebase | Google Cloud SQL | MongoDB Atlas | | Cost | 93.0 | 52,6 | 91,2 | 100 | | Training Days | 53 | 50 | 100 | 60 | | Functionality | 94,7 | 89.5 | 89.5 | 100 |     Normalized Tool Graph |
| Which tool has been selected? Why?  For our project Smart Home Lightning System, Google Cloud SQL is a good choice. It has great features with a score of 89.5, making it a strong option for managing databases. However, it is more expensive, costing 91.2, which is higher than Firebase but lower than MongoDB Atlas. The learning time is also the longest at 30 days, so it will take more time to learn compared to other tools like Firebase, which only takes 15 days. Even with the higher cost and longer learning time, Google Cloud SQL is a good fit for our needs, especially if we need a reliable and flexible database in the cloud. |

|  |
| --- |
| SOFTWARE TOOLS FOR TASK 3: USER BEHAVIOR TRACKING AND PERSONALIZATION |
| Tool Cost/Training/Functionality Data   |  |  |  |  |  | | --- | --- | --- | --- | --- | | Tool | Google Cloud AI | Azure AI Services | AWS AI/ML | IBM Watson AI | | Cost | 3,800 $ | 6,750 $ | 3,500 $ | 4,200 $ | | Training Days | 12 | 21 | 14 | 18 | | Functionality | 87 | 90 | 85 | 88 |   Normalized Cost/Training/Functionality Data   |  |  |  |  |  | | --- | --- | --- | --- | --- | | Tool | Google Cloud AI | Azure AI Services | AWS AI/ML | IBM Watson AI | | Cost | 56.3 | 100 | 51.9 | 62.2 | | Training Days | 85.7 | 100 | 80.0 | 85.7 | | Functionality | 96.6 | 100 | 94.4 | 97.7 |   Normalized Tool Graph |
| Which tool has been selected? Why?  Google Cloud AI stands out as a compelling choice due to its balanced trade-offs between cost, training efficiency, and functionality. While offering nearly the same functionality as Azure AI Services, Google Cloud AI achieves this at nearly half the cost, making it a more economical solution. Additionally, it requires fewer training days, ensuring a faster deployment and reduced time-to-market for AI models. This makes Google Cloud AI a strong contender for businesses seeking an optimal balance between affordability, efficiency, and performance. |

|  |
| --- |
| SOFTWARE TOOLS FOR TASK 4: SMART LIGHT SYSTEM INTEGRATION |
| Tool Cost/Training/Functionality Data   |  |  |  |  |  | | --- | --- | --- | --- | --- | | Tool | Luminair 4 | Lightshow’s The Force | Home  Assistant | Homey  (Smart Hub) | | Cost | 100 $ (Cost Per Unit) | 1100 $  (Cost Per Unit) | 0 $  (Open Source) | 300 $  (Cost Per Unit) | | Training Days | 30 | 10 | 20 | 15 | | Functionality | 50 | 85 | 90 | 80 |   Normalized Cost/Training/Functionality Data   |  |  |  |  |  | | --- | --- | --- | --- | --- | | Tool | Luminair 4 | Lightshow’s The Force | Home  Assistant | Homey  (Smart Hub) | | Cost | 9,1 | 100 | 0 | 27,3 | | Training Days | 100 | 33,3 | 66,7 | 50 | | Functionality | 55,6 | 94,4 | 100 | 88,9 |   Normalized Tool Graph |
| Which tool has been selected? Why?  For this task we considered AI based software to control lightning in user’s houses. When we compare the costs, most expensive tool is Lightshow’s The Force with 1100$ per unit and cheapest is Home Assistant which is an open-source software. When we consider the functionalities less functional tool is Luminar 4 which is exclusively to IOS also lacks some of the functions system requires. Most functional tool is Home Assistant because of its open-source nature we can modify it the way system requires. In terms of training time most time-consuming software is Luminair because we must modify the software also must add an android version. Least time-consuming tool is The Force because it already has an AI feature, so the system only needs to connect with the system. By considering every aspect the graph has most reasonable tool is Home Assistant. Because it is open source, can be modified, and need a small amount of time of training. |

|  |
| --- |
| SOFTWARE TOOLS FOR TASK 5: MOBILE APPLICATION DEVELOPMENT |
| Tool Cost/Training/Functionality Data   |  |  |  |  |  | | --- | --- | --- | --- | --- | | Tool | **Flutter** | **React Native** | **Xamarin** | **Ionic** | | Cost | $120/y | $900/y | $240/y | $300/y | | Training Days | 5 | 4 | 6 | 3 | | Functionality | 90 | 85 | 80 | 75 |   Normalized Cost/Training/Functionality Data   |  |  |  |  |  | | --- | --- | --- | --- | --- | | Tool | **Flutter** | **React Native** | **Xamarin** | **Ionic** | | Cost | 13 | 100 | 26 | 33 | | Training Days | 83.3 | 66.7 | 100 | 50 | | Functionality | 100 | 94.4 | 88.9 | 83.3 |   Normalized Tool Graph |
| Which tool has been selected? Why?  Given the constraints of our project, including a limited timeframe, we needed a tool that offered a balance between cost, ease of learning, and functionality. Flutter stood out as the most suitable choice because it has a relatively low cost compared to other options, requires minimal training days, and provides high functionality. While React Native and Xamarin offer competitive functionality, their higher costs and training requirements made them less viable within our time constraints. |