|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Une image contenant Graphique, capture d’écran, Caractère coloré  Description générée automatiquement |  | **PROJET EPOT** | | |
| **CAHIER DES CHARGES** | | | |
| **Auteurs :**  Arnaud LE COSSEC  Axel NOUY | **Date :**  27/05/23 | | **Révision :** v1.0 | **Pages :** 4 |

I – AVANT PROPOS

Le présent document revient sur les choix réalisés pour le projet EPOT. Nous reviendrons sur la constitution du cahier des charges puis sur la manière dont il a été mis en place.

II – CAHIER DES CHARGES

Le projet consiste en la création d'un pot de fleur intelligent et connecté, contrôlé par une application mobile. Il devra être capable d'arroser automatiquement les plantes à partir de la mesure de l'humidité du sol. L'éclairage devra lui aussi être contrôlé en fonction de la luminosité extérieure afin d'accomplir la consommation la plus faible possible. Il devra également être capable de mesurer et d'afficher la température dans l'air. L'alimentation devra être connectée au secteur et si possible sur batterie alimenter via un panneau solaire. Tous ces éléments devront être accessibles et contrôlables sur une application mobile. Celle-ci affichera les mesures des différents capteurs et laissera à l'utilisateur le choix d'une opération manuelle ou automatique de l'arrosage et de l'éclairage avec la possibilité de sélectionner des préréglages parmi les plantes proposées (niveau d'humidité, luminosité).

Le tableau ci-dessus récapitule ces exigences :

|  |  |  |
| --- | --- | --- |
| Fonctionnalités | Ce qui permet de mettre en œuvre ces fonctionnalités | Précision/ Résolution/ Contraintes |
| Arrosage automatique des plantes | → Capteur humidité du sol  → Capteur niveau du réservoir  → Électrovanne / Pompe / Brumisateur | Résolution de humidité au pourcentage près.  Niveau d’eau au pourcentage près  Arrosage lent et précisément contrôlable |
| Éclairage automatique | → Capteur luminosité ambiante  → Lumière réglable (différentes intensités/couleur)  → Consommation la plus faible possible | Capteur doit pouvoir traduire les différents niveaux de luminosités de la journée.  Éclairage doit pouvoir être supporté par l’alimentation |
| Surveillance de l'atmosphère | → Capteur de température | Doit donner une idée de la température  Résolution au dixième de degré  Précision requise faible |
| Surveillance/gestion sur une application smartphone | → Afficher l'état des capteurs (température niveau d’eau, …)  → Réglage sur différents modes (manuel, automatique, type de plante, …)  L’utilisateur peut choisir lui-même les niveaux (manuel) ou choisi parmi des préréglages selon le type de plante  → Communication des informations via Wi-Fi /Bluetooth | Temps de réaction le plus faible possible |
| Alimentation | → Si possible batterie et panneau solaire  → Sinon sur secteur avec transformateur mural | Doit supporter une utilisation prolongée |
| Prototype | → Impression 3D | → Cacher les composants et les circuits imprimés  → Prévoir des aérations |

III – RÉALISATION DU CAHIER DES CHARGES

Cette partie revient les choix, difficultés et solutions trouvés au cours du projet pour mettre en œuvre le cahier des charges.

1. Microcontrôleur

Le module de commande a été construit autour du microcontrôleur ESP32. Le choix de ce microcontrôleur s'est fait sur la base de ses options de connectivités avec la prise en charge du Wi-Fi et du Bluetooth. Le module de commande a également été réalisé avec la protection des composants en tête : Des diodes et des résistances ont été placées à des positions stratégiques pour empêcher les courants inverses et limiter les effets de pic de tension. Les actionneurs (Pompe, LEDs) sont également isolés du microcontrôleur par des optocoupleurs.

Le firmware du microcontrôleur a pour rôle de récupérer les mesures des capteurs, d'effectuer les conversions nécessaires et de mettre à jour la base de données. Dans le même temps, il doit pouvoir récupérer les instructions de l'application mobile, elles aussi stockées dans la base de données, pour contrôler les actionneurs (Pompe et LEDs).  Le Wi-Fi sera donc utilisé pour communiquer avec la base de données sur internet, tandis que le Bluetooth servira seulement pour la configuration du SSID et du mot de passe Wi-Fi.

Le temps de réactivité du microcontrôleur face aux changements dans la base de données est apparu pour comme un facteur important. Le programme demande donc à la base de données de l'avertir dès qu'il y a un changement, ce qui déclenche une interruption pour récupérer la donnée concernée. Le résultat est un délai maximum de 3 secondes entre la commande sur l'application et la réception de cette commande sur le microcontrôleur. La mise à jour des capteurs, quant à elle est moins critique et n'est effectué que toutes les 5 secondes pour économiser de la bande passante.

1. Application mobile

Pour l'application, nous avons décidé de ne pas utiliser MIT APP Inventor, car malgré sa simplicité d'utilisation, il nous semblait que nous ne pourrions jamais faire ce que nous voulions. Nous avons donc effectué des recherches et nous avons décidé de coder l'application sur L'IDE Android studio qui est spécialisé dans la conception d'application pour Android. Pour ce qui est du langage l'application est codée en Kotlin, le langage de programmation préconisé par Google.

L'application dans sa forme actuelle permet :

- À l’utilisateur de choisir la plante qu'il veut utiliser avec des préréglages sur les niveaux d’humidité et d’éclairage désiré.

- De créer soi-même sa fiche de plante s’il ne trouve pas sa plante

- De mettre toutes les informations relatives aux plantes de l’utilisateur sur une base de données afin de pouvoir y accéder de n'importe où.

- D’afficher les valeurs de tous les capteurs de la manière la plus graphique possible

- De choisir entre un mode automatique et un mode manuel pour l'arrosage et pour l'éclairage

- De choisir l'intensité de l'éclairage pour chaque type de LED indépendamment  
- De communiquer en Bluetooth avec l’ESP32 afin de paramétrer le SSID et le mot de passe du Wi-Fi

1. Arrosage des plantes

Nous avons choisi pour l'arrosage de la plante d'utiliser une pompe péristaltique, car elle nous permet de contrôler de manière très précise la quantité d'eau déversée (goutte-à-goutte). Nous avions également testé la possibilité d'utiliser une autre pompe 3VDC, mais le débit était trop élevé pour notre utilisation. Nous faisons fonctionner cette pompe en 5VDC sur les 12V nominaux pour nous permettre de ralentir encore le débit et de l'alimenter avec la même sortie que les LED.

Pour le capteur d'humidité, nous avons choisi un capteur capacitif de chez Adafruit. Ce choix a été dirigé par sa simplicité d'utilisation et sa disponibilité. Un autre facteur intervenant dans le choix de ce capteur est la présence d'un *solder mask* sur les électrodes qui les protège de la corrosion, contrairement à d'autres capteurs résistifs où elles sont exposées.

Pour pouvoir surveiller le niveau d'eau dans le réservoir, nous avons opté pour un capteur de distance à ultrason. Ce capteur, facilement disponible, est aussi utilisé fréquemment pour la mesure des niveaux dans des cuves. Avec la distance mesurée, nous pouvons facilement effectuer la conversion en pourcentage pour obtenir le niveau d'eau de notre réservoir. Nous avons rencontré quelques difficultés avec ce capteur au cours des différents tests que nous avons effectués. Les parois étroites du réservoir sont propices à l'écho des ondes ultrason. Une mesure effectuée pouvait donc varier fortement avec la précédente. Pour pallier ces erreurs nous avons décidé d'effectuer deux mesures et de ne prendre leur résultat que si elle était identique. Cette méthode est possible, car le niveau d'eau n'évolue pas rapidement.

1. Éclairage

Pour la gestion de l'éclairage nous avons choisi de partir sur 2 types de LED : des LEDs blanches couvrant tout le spectre visuel et des LEDs UV étant donné que certaines plantes en ont besoin pour pousser, mais également si certaines plantes ne sont pas exposées au UV lorsqu'elles sont jeunes elles ne pourront pas être replantées dehors, car elles ne seront pas capables de survivre aux rayons UV du soleil.  Elles seront alimentées en 5V car les, LED blanche ont une chute de tension de 3.2V ce qui rend leur alimentation en 3.3V très complexe.

Pour le capteur de luminosité ambiante, nous avions d'abord porté notre choix sur un phototransistor, car nous l'avions déjà utilisé sur un autre projet.  Cependant, nous nous sommes vite rendu compte que si nous voulions mesurer une intensité lumineuse et non pas un seuil. Le phototransistor n'était pas le meilleur choix. Nous nous sommes alors tournés vers une photorésistance qui permet de mesurer elle l'intensité lumineuse.  Cette dernière nous permettra de décider si oui ou non, il faut éteindre la lumière l'éclairage.

1. Surveillance de la température

Pour la mesure de température, nous nous sommes orientés vers une thermistance CTN (à coefficient négatif). Ce choix a été guidé par notre expérience avec un tel capteur ainsi que pour son coût très faible. Grâce à ce composant, nous pouvons mesurer la température et l'afficher sur notre application. Cependant, nous nous sommes par la suite rendu compte de la faible linéarité du convertisseur analogique-numérique de l'ESP32, résultant en une mesure de températures peu précise. Nous avons dans le code appliqué une correction fixe, mais dans le futur, nous pourrions utiliser un capteur numérique tel que le BMP390.

1. Alimentation

Pour l'alimentation, nous avons choisi de partir sur une alimentation linéaire, car nous avons déjà étudié son fonctionnement auparavant. La faible consommation totale soutient ce choix avec une quantité limité d'énergie à dissiper. Nous avions besoin d'une sortie 5V (LEDs, pompe et capteur ultrason) ainsi que d'une sortie 3.3V (ESP32, autres capteurs).  Nous avons donc choisi d'utiliser 2 régulateurs 5V et un régulateur 3.3V (voir PCB alimentation). Après avoir testé notre PCB nous avons remarqué que les régulateurs 5V ne partageaient pas réellement la charge et qu’un des deux régulateurs chauffait bien plus que l'autre. Nous avons donc décidé de séparer la sortie des deux régulateurs 5V, une pour l'alimentation des LEDs et du capteur ultrason et une pour l'alimentation de la pompe. Si nous devions refaire le projet, nous changerions cette alimentation linéaire pour une alimentation à découpage qui est bien plus efficace et qui permet de tirer beaucoup plus de courant en dissipant moins de chaleur.

1. Prototype

Enfin en ce qui concerne notre prototype nous voulions avoir une conception simple et efficace. Nous avons donc décidé de réaliser nos pièces en impression 3D. Cela nous permet de concevoir sur mesure nos pièces afin de servir parfaitement nos besoins. Le choix du matériau s’est porté sur du PLA, à base d’amidon de maïs biodégradable, participant à une démarche écoresponsable.