

L'IHM à réaliser doit répondre à plusieurs contraintes. La plus évidente à identifier est que cette dernière peut-être utilisée par des individus plus ou moins formés. Elle doit donc répondre au besoin de simples plaisanciers ( Simple, facile de compréhension et d'utilisation ) et au besoin du spécialiste ( pour ce dernier , on peut se permettre des contrôles plus poussés )

*Note : les widgets utilisés ne seront pas les widgets utilisés dans la dernière version. Ces derniers ne sont pas que pour donner une idée de l'interface finale.*

Comment gérer cette dualité au sein de l'interface ? Soit, toutes les commandes sont grisées et ne se débloquent qu'après authentification du spécialiste, soit l'interface change entièrement , selon le mode manuel ou automatique.

Nous avons opté pour changer entièrement l'IHM après authentification ( donc séparer totalement interface manuelle et interface automatique ).

La réflexion est la suivante : même si les boutons désactivés , la présence d'option avancées peut paniquer ou perdre l'utilisateur lambda. Ce dernier pourrait même déduire , en voyant tellement d'options, que l'écluse ne peut pas être opérés correctement avec le peu d'options qui lui sont proposé, et que les fonctions désactivés résultent d'un bug logiciel, etc.

L'IHM automatique et manuelle seront donc totalement séparées.

L'autre critère important de l'IHM est le placement du bouton d'alarme. Ce dernier doit être visiblement, facilement activable , mais ne doit pas être pressé par erreur. Nous avons donc opté pour un bouton rouge de taille conséquente, à droite de l'IHM ( afin de bien le distinguer de l'environnement de contrôle ). Son activation allume un voyant rouge clignotant en haut à droite de l'interface.

**L'interface automatique** sera réduite le plus possible : deux boutons pour sélectionner le sens de passage, et un pour actionner la procédure. Les « feux » de signalisation de l'écluse seront représenté également dans l'IHM et reproduiront le comportement de feux externes ( la corrélation entre les feux virtuels et les feux externes apportant en plus une confirmation à l'utilisateur que la procédure s'est bien déroulé comme prévu , et lui important l'information directement dans la cabane de contrôle ).

**L'interface manuelle** sera affichée uniquement après authentification ( Les widgets nécessaires à cette interface sera invisible sinon ). Pour conserver la logique de l'interface utilisateur, nous mettons les boutons gérant les équipement en aval à gauche et en amont à droite. Pour chaque élément, un voyant indiquera si il est en panne ou non, permettant d'identifier rapidement l'origine du problème.

Au niveau de l'implémentation de l'interface et de la simulation, on peut remarquer une similarité entre les vannes et les portes.

- fonction d'ouverture et de fermeture
- doit capter un signal d'urgence et y réagir
- doit signaler son état
- doit réagir à une panne

On peut donc créer une interface *Equipement* , implémenté par *Vanne* et *Portes*.