

# Documentation fonctionnelle – Onboarding multi-étapes « Apprenant » (BlissLearn)

**Objectif du parcours :** offrir aux nouveaux apprenants de BlissLearn une expérience d'onboarding progressive, qui collecte les informations clés pour alimenter le moteur de recommandation de cours. Le formulaire multi-étapes simplifie la saisie en guidant l'utilisateur étape par étape (progression visuelle, une question à la fois), et fournit un retour immédiat afin de réduire les erreurs et d'optimiser la qualité des données récoltées <sup>1</sup> <sup>2</sup> . Au final, l'utilisateur dispose de recommandations de cours personnalisées selon ses réponses.

## Parcours utilisateur – étapes détaillées

1. **Objectifs d'apprentissage (Step 1) :** question ouverte ou choix dans une liste de catégories. Par exemple, « Quel est votre principal objectif d'apprentissage? (Ex. : « Devenir développeur web », « Maîtriser la cybersécurité », etc.) ». L'utilisateur peut sélectionner un ou plusieurs objectifs prédéfinis ou écrire le sien.
2. **Domaines d'intérêt (Step 2) :** multi-sélection parmi des domaines (ex. Intelligence artificielle, Marketing digital, UI/UX, ...). Présenter les domaines sous forme de cases à cocher pour faciliter la sélection.
3. **Niveau de compétence actuel (Step 3) :** choix unique (boutons radio) pour « Débutant / Intermédiaire / Avancé ». Cette question est peu contraignante, donc placée en début de parcours pour maximiser la progression <sup>3</sup> .
4. **Disponibilité horaire (Step 4) :** saisie du nombre d'heures par semaine (nombre, slider ou champ numérique). Par exemple « Combien d'heures par semaine pouvez-vous consacrer à votre apprentissage? ». Indiquer les plages ou utiliser un curseur pour rendre le choix simple.
5. **Préférences de plateformes (facultatif, Step 5) :** cases à cocher pour les plateformes préférées (Udemy, Coursera, edX, etc.). Cette étape est optionnelle (notée « facultatif » en libellé) pour ceux qui ont déjà des habitudes ou qui veulent laisser libre l'algorithme.
6. **Autres préférences de cours (Step 6) :** regroupe plusieurs sous-questions brèves, par exemple :
7. **Format de cours préféré :** choix parmi « Vidéo, Texte/Slides, Interactif (quiz), autre » (cases à cocher ou liste).
8. **Durée idéale d'un cours :** ex. choix « < 2h, 2-5h, 5-10h, > 10h » (boutons radio ou liste déroulante).
9. **Langue préférée :** liste déroulante ou cases (Français, Anglais, Espagnol, etc.). Ces champs aident à affiner les suggestions (par exemple, "[Quels sont vos préférences de format pédagogique?] <sup>4</sup> ").
10. **Récapitulatif & soumission (Step 7) :** affichage d'un résumé des réponses pour vérification, avec possibilité de revenir modifier chaque étape. Bouton final « Voir mes recommandations » ou similaire.

Chaque étape inclut un en-tête clair (p. ex. « Objectif d'apprentissage », « Niveau actuel », etc.) et un indicateur de progression (barre ou « Étape X/7 ») pour rassurer l'utilisateur qu'il avance dans le formulaire <sup>1</sup> . Les champs essentiels sont marqués « \* obligatoire » (ex. objectifs, domaines, niveau, disponibilité)

tandis que les champs facultatifs sont explicitement labellisés (ex. « Plateformes préférées (optionnel) » <sup>5</sup> ). Les boutons de navigation « Retour » et « Suivant » (ou « Continuer ») sont toujours visibles, le bouton de la dernière étape étant libellé selon l'avantage final (ex. « Afficher mes recommandations » plutôt que « Soumettre », pour plus de clarté <sup>6</sup> ).

#### Schéma des étapes (résumé) :

- **Step 1** : Objectifs d'apprentissage (texte ou choix)
- **Step 2** : Domaines d'intérêt (multi-cases)
- **Step 3** : Niveau de compétence (radios : Débutant/Intermédiaire/Avancé)
- **Step 4** : Disponibilité (nombre d'heures/semaine)
- **Step 5** : Plateformes préférées (Udemy, Coursera, edX...) – *facultatif*
- **Step 6** : Préférences de cours (format, durée, langue)
- **Step 7** : Récapitulatif et validation finale

## Recommandations UX et bonnes pratiques

- **Progression visuelle claire** : utiliser une barre de progression ou un indicateur d'étape (« Étape X sur Y »). Cela motive l'utilisateur à continuer et lui donne une idée du temps restant <sup>1</sup> . Par exemple, une barre colorée en haut de page ou un pas-à-pas numéroté (1→7).
- **Un seul champ par ligne, labels explicites** : structurer chaque étape en une colonne simple (pas de mise en page à plusieurs colonnes) pour éviter toute confusion <sup>7</sup> . Afficher le label de chaque champ **au-dessus** de la zone de saisie, et non comme placeholder temporaire, car les placeholders disparaissent à la frappe et sont moins accessibles <sup>8</sup> . Exemple : label « Niveau de compétence actuel » en gras au-dessus de trois boutons radio.
- **Questions claires et ton engageant** : formuler chaque question de façon simple, courte et positive. Utiliser « vous » pour personnaliser (ex. « Quel est VOTRE objectif principal aujourd'hui ? »). Donner des exemples pour inspirer (« p. ex. devenir développeur web, maîtriser le marketing digital... » <sup>9</sup> ). Employer un ton encourageant : « Choisissez les domaines qui vous motivent afin de recevoir des recommandations personnalisées ! ».
- **Validation instantanée (validation douce)** : mettre en place une validation **en ligne** (client-side) lorsque l'utilisateur quitte un champ ou passe à l'étape suivante, plutôt qu'à chaque frappe <sup>2</sup> <sup>10</sup> . Cela évite de l'alerter trop tôt et de le frustrer. Par exemple, si le champ « heures par semaine » attend un nombre, un message d'erreur ne s'affiche qu'une fois que le champ est rempli (ou au minimum sur « blur »), ou lors du clic sur « Suivant » <sup>11</sup> .
- **Messages d'erreur clairs et visibles** : si la saisie est invalide (format incorrect, champ vide), afficher un message explicite sous le champ concerné <sup>11</sup> <sup>12</sup> . Plutôt que « Erreur », préférer un message du type « Ce champ est obligatoire » ou « Veuillez entrer un nombre valide ». Insister sur la solution (ex. « Merci de renseigner votre disponibilité en heures par semaine (chiffre) » <sup>12</sup> ). Les champs invalides peuvent être bordés de rouge et le bouton « Suivant » désactivé tant que les erreurs ne sont pas corrigées <sup>13</sup> <sup>11</sup> .
- **CTA explicites** : les boutons d'action doivent clairement indiquer l'étape ou l'action (« Suivant », « Étape suivante », enfin « Afficher mes recommandations ») et être visuellement marqués (couleur, taille) pour ressembler à des vrais boutons <sup>14</sup> <sup>6</sup> . Éviter les libellés vagues comme « Envoyer » ou « Continuer » sans contexte <sup>6</sup> .
- **Retour (back) possible** : offrir un bouton « Précédent » pour chaque étape (sauf la première) afin que l'utilisateur puisse corriger ses réponses sans tout recommencer. Chaque clic sur « Suivant » peut

sauvegarder provisoirement les données du formulaire (en localStorage ou en session) pour éviter toute perte en cas de fermeture accidentelle.

En somme, l'expérience doit être guidée et rassurante <sup>1</sup> : l'utilisateur reste « en contrôle » tout au long du parcours, ce qui minimise l'abandon <sup>1</sup>. Les questions faciles sont placées en début (pour éliminer la barrière psychologique) <sup>3</sup>, et la complexité des informations à fournir est introduite progressivement. Un design épuré, des instructions brèves et des exemples aident à la compréhension.

## Exemples de libellés / wording engageant

- **Objectifs d'apprentissage** : « Quel(s) objectif(s) visez-vous en suivant des cours sur BlissLearn ? Par exemple : « Devenir développeur web », « Améliorer mes compétences en UX/UI », « Se certifier en cybersécurité »... Rédigez votre propre objectif ou choisissez parmi la liste ».
- **Domaines d'intérêt** : « Dans quels domaines souhaitez-vous vous former ? Cochez les sujets qui vous intéressent (p. ex. Intelligence Artificielle, Marketing digital, Data science, ...). »
- **Niveau de compétence** : « Quel est votre niveau actuel dans ces domaines ? (choisissez « Débutant », « Intermédiaire » ou « Avancé »). Ne vous inquiétez pas, même les débutants progressent rapidement ! »
- **Disponibilité horaire** : « Combien d'heures par semaine pouvez-vous consacrer à votre apprentissage en ligne ? » (expliquer « Par exemple : 3h/sem. », prévoir un champ numérique ou curseur).
- **Plateformes préférées (facultatif)** : « Sur quelles plateformes suivez-vous habituellement vos cours ? (Udemy, Coursera, edX...). Cette info nous aide, mais vous pouvez passer cette étape si vous n'avez pas de préférence. »
- **Format de cours** : « Quel format de cours préférez-vous ? (p. ex. vidéos, lectures textuelles, quiz interactifs) <sup>4</sup>. »
- **Durée idéale** : « Quelle est la durée idéale pour un cours selon vous ? Moins de 2h, 2-5h, 5-10h, plus long... ».
- **Langue préférée** : « Dans quelle langue préférez-vous suivre vos cours ? (Français, Anglais, Espagnol, etc.) ».

Ces formulations utilisent un ton positif et motivant, soulignant l'avantage personnel de chaque réponse (ex. « poursuivez vos projets... », « nous adapterons vos recommandations... »). Elles emploient la deuxième personne (« vous ») pour impliquer directement l'apprenant.

## Gestion des données et validation

- **Stockage temporaire** : les réponses peuvent être sauvegardées au fur et à mesure (par exemple dans l'état de l'application ou dans le navigateur) pour garantir qu'un utilisateur peut reprendre le formulaire s'il interrompt le processus. Lors du clic sur « Suivant », on enregistre chaque étape en local ou via l'API backend en prévisualisation, sans validation finale.
- **Validation des champs** : certaines données doivent respecter un format (nombre d'heures, adresse email si on en demandait une, etc.). Utiliser les attributs HTML (type, `required`, expressions régulières) pour la validation côté client et JavaScript pour le feedback instantané <sup>2</sup> <sup>10</sup>. Toutefois, **toujours ré-contrôler** les données côté serveur avant de les stocker définitivement <sup>10</sup>. En effet, la validation client améliore l'UX mais ne remplace pas la sécurité serveur : il faut vérifier les mêmes contraintes en backend <sup>10</sup>.

- **Champs obligatoires clairement signalés** : tous les champs nécessaires au système de recommandation (objectifs, domaines, niveau, disponibilité) sont marqués par un astérisque ou un label « \*Obligatoire » <sup>5</sup> . Les champs optionnels sont indiqués (« facultatif »). Ceci évite les soumissions incomplètes.
- **Erreurs et retours** : en cas de champ manquant ou mal formaté, afficher immédiatement un message explicatif (par exemple « Ce champ est obligatoire » ou « Entrez un nombre valide »), près du champ concerné <sup>12</sup> <sup>11</sup> . Utiliser des couleurs et des icônes familières (texte rouge, point d'exclamation) pour attirer l'attention. Ne pas envoyer d'erreur générique – fournir toujours la solution (« Merci de saisir un nombre entier pour les heures par semaine ») <sup>12</sup> <sup>11</sup> .
- **Retour d'information positif** : lorsque l'utilisateur corrige une erreur, indiquer la réussite (par exemple, coche verte). Cela crée un sentiment de progression positive. Ne pas forcer la correction d'emblée ; on peut laisser l'utilisateur compléter le formulaire et afficher un récapitulatif des erreurs restantes avant la soumission finale <sup>13</sup> .
- **Sécurité et conformité** : bien que l'on collecte des préférences pédagogiques, il est recommandé de rappeler la confidentialité des données (RGPD) si nécessaire. Par exemple, un lien vers la politique de confidentialité ou une note indiquant que les données sont utilisées uniquement pour améliorer l'apprentissage de l'utilisateur.

Ces mesures garantissent la fiabilité et la complétude des données collectées. La validation « douce » (en ligne et au besoin au clic suivant) permet de guider l'apprenant sans le bloquer brusquement <sup>2</sup> <sup>10</sup> . Comme le note Microsoft, la combinaison de la validation client (rapide, conviviale) et serveur (sécurité) offre la meilleure expérience utilisateur <sup>10</sup> .

**Sources** : Ce document s'appuie sur les meilleures pratiques UX pour formulaires multi-étapes <sup>1</sup> <sup>2</sup> , ainsi que sur des recommandations en rédaction et design de formulaires <sup>8</sup> <sup>12</sup> <sup>10</sup> <sup>15</sup> , adaptées au contexte BlissLearn.

---

<sup>1</sup> <sup>2</sup> <sup>3</sup> <sup>5</sup> <sup>11</sup> <sup>14</sup> Les meilleures pratiques UX à suivre lors de la conception d'un formulaire à plusieurs étapes - Growform Multi Step Form Builder

<https://www.growform.co/fr/les-meilleures-pratiques-ux-a-suivre-lors-de-la-conception-dun-formulaire-a-plusieurs-etapes/>

<sup>4</sup> <sup>9</sup> <sup>15</sup> 30 questions incontournables pour vos sondages d'évaluation des formations (2025)

<https://fr.surveymonkey.com/mp/training-survey/>

<sup>6</sup> <sup>7</sup> <sup>8</sup> <sup>12</sup> Les bonnes pratiques pour les formulaires mobiles ! ??

<https://www.media.thiga.co/les-bonnes-pratiques-pour-les-formulaires-mobiles>

<sup>10</sup> Validation côté client (validation dans les couches de présentation) - .NET | Microsoft Learn

<https://learn.microsoft.com/fr-fr/dotnet/architecture/microservices/microservice-ddd-cqrs-patterns/client-side-validation>

<sup>13</sup> validation - Validate on blur or keypress? - User Experience Stack Exchange

<https://ux.stackexchange.com/questions/48268/validate-on-blur-or-keypress>