IMPROVING THE ONBOARDING USER EXPERIENCE IN MOOCS

Jan Renz, Thomas Staubitz, Jaqueline Pollack, Christoph Meinel

Hasso Plattner Institute (GERMANY)

Abstract

A lot of research has been done in maintaining and increasing the intrinsic motivation of learners within online courses. Especially in Massive Open Online Courses (MOOCs), where, due to the high amount of course participants, personal supervision cannot be offered this is crucial to avoid high dropout rates. The users' journey into the world of online learning begins earlier, however. In an ideal situation, prior to starting a course, users would make themselves comfortable with the offerings of the platform. In an even more ideal world, they would be assisted and supported by the platform during this process.

Registration, enrollment, understanding the basic structure and functionality of the MOOC-platform, the handling of blank states (e.g. a progress page right at the beginning of the course when there is no user progress available yet) are just a few obstacles the freshman user has to master before she can start with studying the actual topics of the course.

The paper at hand first classifies the different onboarding mechanisms and insights from outside the e-learning scope.

In the following, current approaches of major MOOC platforms are described and evaluated. Based on the evaluation results, a best-practice guide including several suggestions is presented for MOOC-providers, which could also be applied to other types of Learning Management Systems (LMS).

Keywords: gamification, onboarding, on-boarding, mooc, e-learning

1 INTRODUCTION

openHPI is Germany's largest Massive Open Online Course (MOOC) platform with a specialization in ICT engineering. Run by the Hasso Plattner Institute (HPI) in Potsdam, it has offered nine courses on various ICT topics since September 2012—hosting between 5,000 and 17,000 enrolled users per course. Only basic onboarding features have been included in the first version of the openHPI platform (v1). This paper's purpose is to identify best practices and suggestions for a better onboarding that could be used by all MOOC and Learning Management Systems (LMS) providers.

1.1 Onboarding

The term onboarding is mostly used in the business context:

> "Onboarding, also known as organizational socialization, refers to the mechanism through which new employees acquire the necessary knowledge, skills, and behaviors to become effective organizational members and insiders" [1]

In this paper a different meaning of the term will be used. Game designers are facing the task of onboarding a user to complex scenarios quite often. According to the usage of the word in the context of gamification(see [2])—"the use of video game elements in non-gaming systems to improve user experience (UX) and user engagement" [3], it can be defined as the slow increasement of the system's complexity, delivering positive reinforcement to avoid early fails and get to know the users[2]

In this paper onboarding is understood as the sum of methods and elements helping a new user to become familiar with a digital product—be it a web-based application, mobile app, or another digital application. By providing onboarding mechanisms, users will be enabled to smoothly pass into the efficient usage of the digital product.

A user's first minutes in a new digital environment are fundamental for his user experience. A good onboarding process should deliver a smooth and frictionless experience. In the book *Gamification by design*, Zichermann and Cunningham are suggesting the following key tasks of onboarding:

They suggest a user path that starts with **actions**, delivers **early rewards** and leads to a **registration** of the user. In this paper a more differentiated review of the user onboarding based on the requirements of MOOC platforms is developed.

Onboarding can be described as the first of three phases in supporting a users activity on a MOOC platform:

Phase 1: Onboarding—Helping the user to become an effective user of the system

Phase 2: Help and Support—Providing support and Motivation to the user while using the platform (Examples: Gamification, Course communication)

Phase 3: Re-engagement—Reactivating users who haven't been active within a course or the platform for some time (Example: "Haven't seen you for a while" email).

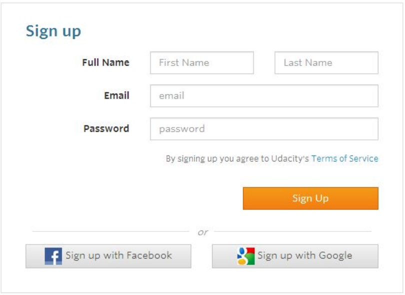
2 ELEMENTS OF ONBOARDING

2.1 Login and Registration

Many websites and Internet applications require the user to register and sign up before she can access the website. Basically there are two steps:

The **registration** introduces the new user to the platform. The Minimum of required data are user name—often the email address is used for that purpose—and a password. Often a validation of the provided email is required to complete the registration. If so the user has to wait for the confirmation email to arrive, needs to find that mail (it might have been sorted into a Junk folder or into some smart folder like the notification folder in Gmail), and click the provided link that will confirm that she is the owner of that email address. Once a user is registered on the platform she is able to **log in** to the system.

Some platforms also enable the users to login with an existing, already confirmed account created on an other "trusted" platform, in general a social network such as Facebook or Google+ and the likes—a so called social login[1]. At openHPI over 50% of the users have a Facebook, Google+, etc. account [4]. A **social login** reduces the amount of required user interaction and thereby increases speed and comfort of the registration process. The user is redirected to the login provider, where she is asked to allow sharing the requested information with the MOOC platform. No need to search for confirmation emails over and over again. In the example illustration (Fig. 2) the MOOC platform requested basic profile info and the list of friends). It is recommended to only request the mandatory information for the purpose of registration and to ask for an extension of these permissions later only if more information is needed (In this example only basic profile info would be requested). The list of users is only required if the user wants to invite her friends or if she wants to connect to her friends within the platform).



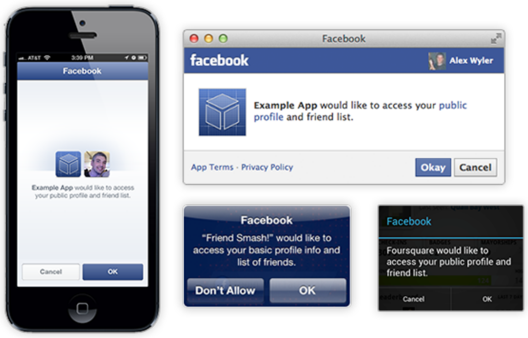
Sign up

Full Name: First Name | Last Name

Email: email

Password: password

By signing up you agree to Udacity's Terms of Service



Facebook

Example App would like to access your public profile and friend list.

Facebook

facebook Alex Wyler

Example App would like to access your public profile and friend list.

Facebook

"Friend Smash!" would like to access your basic profile info and list of friends.

Facebook

Foursquare would like to access your public profile and friend list.

Figure 1 (left): Sign-Up/Registration form with social login button, Figure 2 (right): A social login asks for the users permission, Screenshots from different environments

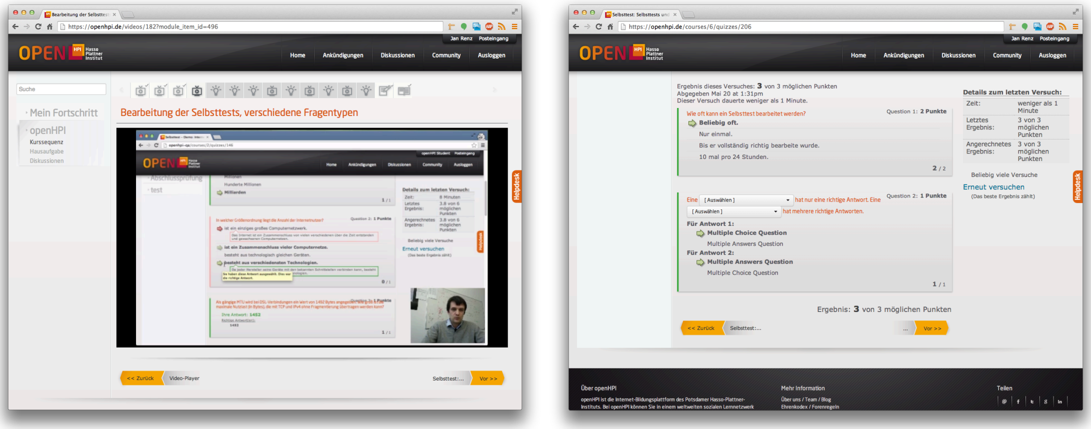
A social login reduces the registration and login process to a minimum of two clicks. No data must be re-entered. Registration and login are handled within one simple step. Social login is available on a

[1] OAuth (and its current version OAuth2) can be considered the technical standard for this social logins. On https://oauth.io a list of popular social login providers can be found. SAML, a different authentication standard, may be used as well.

broad range of devices and can be used in web applications and native apps. A social login offers an easy and comfortable alternative to the traditional login process.

2.2 Demo Course

In openHPI v1, a dedicated introductional MOOC was offered to introduce the users to the—then new— platform.



Bearbeitung der Selbsttests, verschiedene Fragentypen

https://openhpi.de/videos/182?module_item_id=496

In welcher Größenordnung liegt die Anzahl der Internetnutzer?

Question 2: 1 Punkte

ist ein einziges großes Computernetzwerk.

ist ein Zusammenschluss vieler Computernetze.

besteht aus technologisch gleichen Geräten.

besteht aus verschiedensten Technologien.

Als gängige MTU wird bei DSL-Verbindungen ein Wert von 1492 Bytes angegeben. Wie groß ist die maximale Nutzlast (in Bytes), die mit TCP und IPv4 ohne Fragmentierung übertragen werden kann?

Ihre Antwort: 1452

Richtige Antwort(en): 1452

https://openhpi.de/courses/6/quizzes/206

Ergebnis dieses Versuches: 3 von 3 möglichen Punkten
Abgegeben Mai 20 at 1:31pm
Dieser Versuch dauerte weniger als 1 Minute.

Wie oft kann ein Selbsttest bearbeitet werden?

Question 1: 2 Punkte

Beliebig oft.

Nur einmal.

Bis er vollständig richtig bearbeite wurde.

10 mal pro 24 Stunden.

2 / 2

Eine [Auswählen] hat nur eine richtige Antwort. Eine [Auswählen] hat mehrere richtige Antworten.

Question 2: 1 Punkte

Für Antwort 1:
Multiple Choice Question
Multiple Answers Question

Für Antwort 2:
Multiple Answers Question
Multiple Choice Question

1 / 1

Ergebnis: 3 von 3 möglichen Punkten

Details zum letzten Versuch:

Zeit:	weniger als 1 Minute
Letztes Ergebnis:	3 von 3 möglichen Punkten
Angerechnetes Ergebnis:	3 von 3 möglichen Punkten

Beliebig viele Versuche

Erneut versuchen
(Das beste Ergebnis zählt)

Über openHPI

openHPI ist die Internet-Bildungsplattform des Potsdamer Hasso-Plattner-Instituts. Bei openHPI können Sie in einem weltweiten sozialen Lernnetzwerk

Figure 3 (left) & 4 (right): Two screenshots of the demo course in openHPI v1.

Provided in the same format as the “real” courses, the demo course offered an introduction to topics like

- How to navigate
- How to use the dual screen video player
- How to read the progress
- Requirements to receive a certificate

In Figure 3 a video explaining the quiz functionality is shown. In Figure 4 a quiz about the quiz functionality is shown. Its notable that the quiz layouz in Figure 4 differs from the one in the video tutorial cause the quiz layout has been improved while the tutorial was not rerecorded.

5525 users enrolled in this demo course. 24.3% of these users completed it and received a “demo” certificate. This success rate is comparable to the success rates of other openHPI courses, which have a 6 week range, “real” topics, are way more ambitious, and have a much higher workload [5]. Therefore, it has to be considered as alarmingly low.

The demo course approach has some drawbacks:

The user path to the course is stretched. New users will have to spend approximately 10-20 minutes before they even get in contact with the actual, relevant course content. The user has to learn the platform detached from the context of the course. This assumes that the users have a natural interest in the functionality of the platform itself. This assumption is not valid for the large majority of MOOC platform users. Their primary focus should be on the course contents—including both: course materials as provided by the teaching team as well as user generated content in the forum or learning groups.

2.3 Platform tutorial in the course context

In many MOOC courses the lecturer includes some platform explanations in his first lecture. This might be a comfortable solution for users taking their first course on this MOOC platform, but it will bore those users who already visited a course on that platform or are familiar with the platform by exploring. So In-course platform tutorials should be brief and easy to skip. If a users not watching a In-course platform tutorial there should be no negative impact on the course progress.

2.4 Public Sections

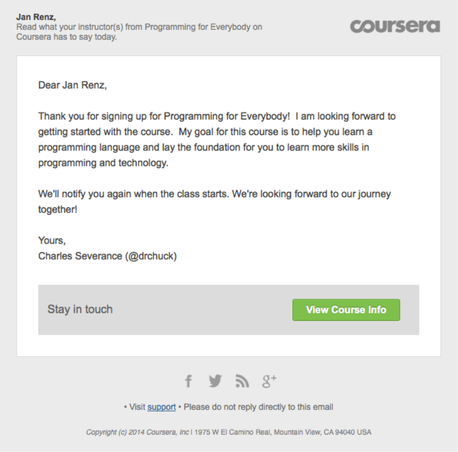
A simple approach to allow users exploring both: the functionalities of the MOOC platform and the contents of a specific course could be to make certain sections of a course available in guest mode. However, even in this simple approach restrictions apply, for example only assignments that allow unlimited attempts can be used within such a section. Two different approaches to grant public access to a section exist:

a) A section may be public to not enrolled users (who are registered on the platform)**.** This allows users to take their time to decide if they want to enroll to that course.
b) A section may be public to users that are not logged in on the system.

2.5 Welcome Mails

Two different kinds of welcome mails have been identified. The first one is the **platform welcome mail**, which is sent out to the user after registering. The second one is the **course welcome mail**, which will be sent out to a user every time she enrolls to a course.

In the most frequent registration scenario the user registers to be able to enroll in a course. As a result two mails will be sent to the user. The example welcome mail (see Fig. xx)—taken from Coursera—provides several action points for the user such as enrolling to courses or joining the social communication channels of the MOOC provider. The course welcome mail in Fig. xx states that the course will start soon, without even telling the course start date.



Jan Renz,
Read what your instructor(s) from Programming for Everybody on Coursera has to say today.

Dear Jan Renz,

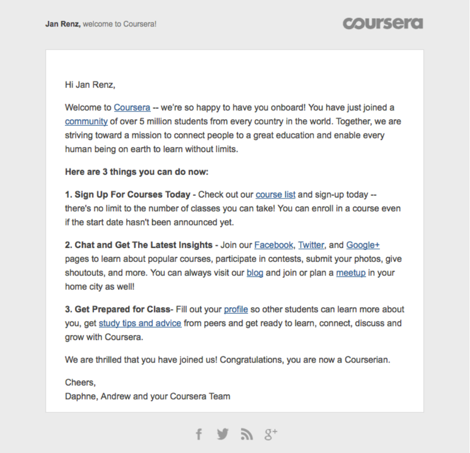
Thank you for signing up for Programming for Everybody! I am looking forward to getting started with the course. My goal for this course is to help you learn a programming language and lay the foundation for you to learn more skills in programming and technology.

We'll notify you again when the class starts. We're looking forward to our journey together!

Yours,
Charles Severance (@drchuck)

• Visit support • Please do not reply directly to this email

Copyright (c) 2014 Coursera, Inc | 1975 W El Camino Real, Mountain View, CA 94040 USA



Jan Renz, welcome to Coursera!

Hi Jan Renz,

Welcome to Coursera -- we're so happy to have you onboard! You have just joined a community of over 5 million students from every country in the world. Together, we are striving toward a mission to connect people to a great education and enable every human being on earth to learn without limits.

Here are 3 things you can do now:

1. Sign Up For Courses Today - Check out our course list and sign-up today -- there's no limit to the number of classes you can take! You can enroll in a course even if the start date hasn't been announced yet.

2. Chat and Get The Latest Insights - Join our Facebook, Twitter, and Google+ pages to learn about popular courses, participate in contests, submit your photos, give shoutouts, and more. You can always visit our blog and join or plan a meetup in your home city as well!

3. Get Prepared for Class- Fill out your profile so other students can learn more about you, get study tips and advice from peers and get ready to learn, connect, discuss and grow with Coursera.

We are thrilled that you have joined us! Congratulations, you are now a Courserian.

Cheers,
Daphne, Andrew and your Coursera Team

Figure 5 (left): course welcome mail, Figure 6 (right): platform welcome mail

As welcome mails received by the users' devices will probably lead to notifications on her PC or smartphone, this will interrupt the onboarding process, which—if well designed—would include the very same action points. On the other side emails have the advantage to be available offline. So, if a user searches his computer for e.g. the course title or the platform name she will find these mails. This offers the opportunity to over the user information she might want to access without opening the course page (due dates, important links).

3 USAGE OF ON-BOARDING BY MAJOR MOOC PROVIDERS

In this chapter the authors of this paper evaluated 6 major MOOC providers and their usage of on-boarding instruments.

edx is a non-profit company governed by MIT and Harvard, the platform is open-sourced. It was established in fall 2012 and has more than 2.1 million users. Several other universities or companies also use the same software with their own brand, usually labeled with a small 'x' as a suffix (e.g. HarvardX, BerkeleyX). **Coursera**, being one of the first MOOC providers, started in fall 2011 and has more than 7.1 million users. **Udacity** is a for-profit organization, which released its first courses in February 2012 and features over 1.6 million users. Besides these three US-based MOOC providers, three European providers have been evaluated:

- **FutureLearn**—one of the new companies who started in September 2013.
- **MiriadaX**—the largest Spanish MOOC provider with 670.000 Users,

- **Iversity—**a commercial German MOOC provider with over 500.000 users as of February 2014[2]).

All together this sample group reaches more than 10 million users. As shown in Table 1 all MOOC providers require some form of registration and log in before the platform can be used. Usage prior to login is very limited. A social login eases registration at some platforms. None of the MOOC providers in the sample group provides a real smooth onboarding experience like it can be found on some digital products outside the MOOC scope.

Table On-Boarding Features

	edX	**Udacity**	**Coursera**	**Futurelearn**	**MiriadaX**	**Iversity**
URL	edx.org	udacity.org	coursera.com	futurelearn.com	miriadax.net	iversity.org
Tasks without registration	Search for courses	Search for courses	Search for courses	Search for courses	Search for courses	Search for courses
Tasks without enrollment	-	Demosections	-	-	-	-
Data needed for registration	Email, password, nickname, fullname	Email, password, name or Facebook /google+	Email, password, name	Email, password, name	Email, name	Email, password, name or Facebook
Mail with registration link	yes	yes	yes	no	no	yes
Welcome Mail	no	no	no	yes	yes (incl. password)	no
Course welcome Mail	no	yes	yes	yes	no	
Demo Video or Course	Yes	No	Demovideo	Demovideo in course	No	No
FAQ / Support-Contact	Yes / Yes	Yes / Yes	Yes / Yes	Yes / Yes	Yes / No	Yes / Yes

Table 1: On-boarding features of major MOOC providers. Some functionalities might be available in some courses and therefore have not been detected.

4 SUGGESTED IMPROVEMENTS

4.1 Use common UI UX patterns

Apple followed a very strict policy after opening its iPhone device for third party apps. App developers had to follow a detailed design guideline, describing how to solve certain design tasks. Later both Android and Windows also published similar documents[3]. As a result many developers followed this design guidelines. This resulted in a large amount of apps sharing the same UI-patterns. If a user

[2] http://techcrunch.com/2014/02/19/iversity-500k-sign-ups/

[3] See: https://developer.apple.com/library/ios/documentation/userexperience/conceptual/MobileHIG/index.html, https://developer.android.com/design/index.html, http://dev.windowsphone.com/design/library

knew how to interact with e.g. Apples Mail app, he would be able to interact with all apps following the same UI-pattern.

Depending on the overall web-experience of the users, many of the functionalities require zero explanations. It can be assumed that nearly every user knows how pagination works. Naturally, newer and less common UI/UX patterns and platform specific functionalities will demand for more explanations.

4.2 Lazy Login/Registration

For a long time platform providers only considered two states. Either a user is logged in or not. Early registration, however, requiring the provision of personal data, might stop users from further exploring a MOOC platform. In ecommerce and online shops, where understanding the users behavior and expectations is essential and can be easily measured in terms of revenue, the lazy login/registration pattern is well know and established. If the user enters a web shop, e.g. amazon.com she is immediately able to use a great deal of the provided features and functionalities. Only when the shops business logic really requires to know the user’s identity, e.g. for checking out the shopping basket, she has to login or register. This usually happens as late as possible. It does not mean that an early login is not possible. In fact, an early login is promoted by awarding the user with additional features, such as personalized recommendations and offers. During the process of lazy login/registration, previously collected data such as the items in the shopping cart need to be persisted and should still be available to the user when she has logged in or registered.

This pattern applied to a MOOC scenario would allow a user to enter a course, watch the videos, and take self-tests without signing up. Some actions, especially those including the process of publishing content on the platform or taking actions that would result in acquiring points that are relevant for a certificate (such as posting in a forum or taking an exam) still should require the user to log in first.

Some issues of this pattern should be taken into consideration before implementing it:

a) The amount of registered users will decrease.

If users who want to peek only or are insecure if they want to join a course are not required to sign up or to enroll to the course, the total amount of platform users and course participants will be lower than before. In turn, those users who decided to sign up and enroll, are more dedicated towards the platform and/or the courses. It can be assumed that this would lead to better completion rates and a higher percentage of active users on the platform. Or in other words: Quality in exchange for quantity [5].

b) Users might lose data

Generally, users who are browsing the platform and courses in a pre-login state will generate data and are able to transfer this data after signing up or logging in. For example if a user already watched some videos and took some self-tests this progress-related data must be be part of his course progress after log in. To be able to provide the relation between the logged in user and the guest users section the guest user session will be stored via a cookie. If the user visits the platform again, the platform can detect this cookie and restore the matching guest users session. The cookie approach has some major drawbacks: Cookies are bound to a certain browser. So if the user switches to another device, the system will not be able to restore the guest users session containing the progress. Also some browsers (based on their settings) might delete the cookies after every restart. While it might be possible to work around these issues by trying to use other techniques to store the cookie (such as Local Store or Browser-Fingerprinting) the authors of this paper recommend to encourage (but not force) the user to register. This could happen by a visual notice describing the benefits of signing up.

Furthermore, a soft limit could be defined, specifying a certain amount of user interactions or data quota. If a soft limit is chosen, this limit should be high enough to enable the user to evaluate the platform and a course without reaching the limit. When the limit is hit, further usage of the platform should be locked, combined with a notice asking the user to register and log in.

4.2.1 Merging Data after login of a already signed up user

If this pattern is applied, the implications of the platform usage by users that are registered but currently not logged in have to be considered as well. This might happen as easily as users will forget to login. In this situation a user will want to be able to login without losing data. This scenario raises some issues: The PC used by one user might have been used by a different user before e.g. mother and daughter. This might be considered as an exotic edge case but these edge cases exist, so a solution is required.

4.2.2 Avoiding data conflicts

This simplified approach of just adding previously generated data assumes that adding data is possible in all cases. In MOOC environments this might not be true all the time. So, a strategy to handle potential conflicts is required:

A simple but powerful solution would be to avoid all user interactions that might lead to conflicting data. Assumed a course consists of videos that can be watched infinite times, several self-tests that can be taken unlimited times and an exam that can only be taken once. While in guest mode, the user should be able to watch the videos and take the self-tests. Taking the exam, however, will request a login.

4.3 Feature by feature

At the end of April 2014 Sony mobile research published a prototype of a smartphone UI for Android called EvolutionUI [6]. In this prototype the smartphone initial state starts with a very simple setup on the phone with 4 apps only and without the possibility for customization. The user must then complete certain tasks to achieve badges and unlock new features, for example she must open every app at least once to unlock the list of all apps installed on the phone.

This approach might is keeping the UI very simple so the user can focus on the offered basic tasks. Applying this approach to the MOOC context is difficult, as it is based on an increasing complexity of the usage of the digital product. The assumption that a smartphone might want to check the four visible apps before checking the list or that the user might check the installed apps before installing new ones are reasonable. But based on the available usage statistics it is hard to identify similar usage patterns for MOOC providers. There might be some areas within a course that might only be needed after some time (progress, certificate download). For example the page where users can download their documents (certificates) could be hidden and enabled only after certain events (for example the issuing of the first document). However this will lead to two disadvantages that need to be considered:

Depending on their current state users will see different user interfaces. One might have less navigation possibilities, which might lead to confusion if compared with screenshots or help texts within the platform. A solution to this issue might be to have the tabs displayed, but disable them. However this will also need to confusion cause it might not be obvious to the user why he cant open that certain page.

While this might be a nice idea, it should always be compared if the empty state pattern described in the following chapter isn't a smarter alternative.

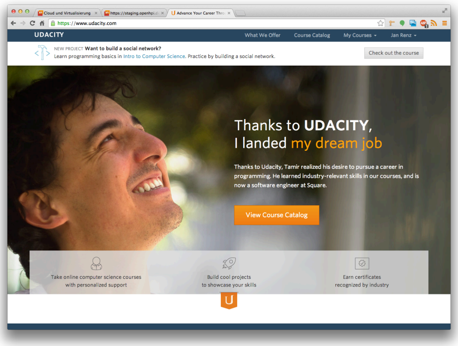
4.4 Handling the blank state

The blank state (sometimes called the empty state) describes the state within a certain page, as long as the data that the page is intended to show is not available. This could be an email inbox before any email arrived. Empty states can be used to communicate to the user what he can expect from this section and what actions might be needed to see data here. For example in the documents section the users could be informed how a certificate will look like and what he needs to do to receive a certificate. She will also be informed that this is the part of the platform where she can download the documents. Blank States can be full pages or parts of a page. Considered there is a sidebar with several contents and one of the boxes is displaying the activity feed of the users friends. If there is no content available the box could encourage the user to connect with friends no to see their activities.

4.5 What to do after re-login

When a user logs in to a MOOC platform again, it can easily be assumed that she wants to resume her work in one of the courses in which she is enrolled. The platform should be able to deal with this most probable use case. The screenshot above shows the Udacity website after login. A large space of the page is used for an internal advertisement. To access his course the user has to access the top menu bar, and select “My courses”, so a drop down will appear with the users current course enrollments. After selecting an item from the courselist she then will be followed to the course.

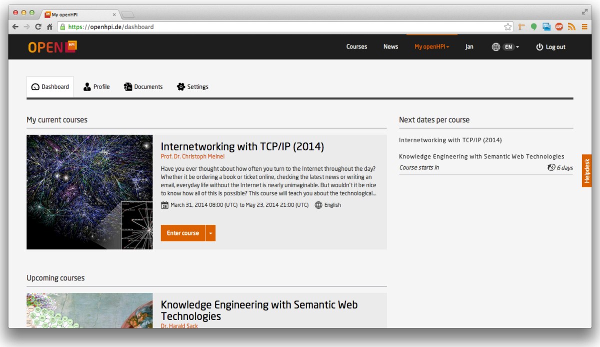
One approach is to directly forward the user to the last visited course. This makes sense if the platform only provides one course at a time. This might also make sense if the user has the possibility to enroll to multiple courses. Because its very difficult if not impossible to predict which course the users wants to visit after her login, the most straight forward strategy would be to forward her to the last course she interacted with.


Cloud und Virtualisierung
https://staging.openhpi.d
Advance Your Career Thro
https://www.udacity.com
UDACITY
What We Offer
Course Catalog
My Courses
Jan Renz
Check out the course
View Course Catalog
Take online computer science courses with personalized support
Build cool projects to showcase your skills
Earn certificates recognized by industry

NEW PROJECT Want to build a social network?
Learn programming basics in Intro to Computer Science. Practice by building a social network.

Thanks to UDACITY, I landed my dream job

Thanks to Udacity, Tamir realized his desire to pursue a career in programming. He learned industry-relevant skills in our courses, and is now a software engineer at Square.



My current courses

Internetworking with TCP/IP (2014)
Prof. Dr. Christoph Meinel
Have you ever thought about how often you turn to the Internet throughout the day? Whether it be ordering a book or ticket online, checking the latest news or writing an email, everyday life without the Internet is nearly unimaginable. But wouldn't it be nice to know how all of this is possible? This course will teach you about the technological...
March 31, 2014 08:00 (UTC) to May 23, 2014 21:00 (UTC) English

Upcoming courses

Knowledge Engineering with Semantic Web Technologies
Dr. Harald Sack

Next dates per course

Internetworking with TCP/IP (2014)

Knowledge Engineering with Semantic Web Technologies
Course starts in 6 days

Figure 6 (left): Udacity Home page (logged-in), Figure 7 (right): openHPI v2 Homepage (logged-in)

openHPI v2 implements a different approach. The screenshot on the right (Figure 7) shows the openHPI dashboard, a user centric page shown after login. It will list all current courses ordered by last interaction. By clicking the "Resume course" button, the user will be forwarded to the last page she accessed within the selected course. In the right sidebar further information that is relevant for the user can be provided, such as approaching deadlines, received messages or outstanding peer reviews. It provides a single point of information, from where the course resume activity is just one click away.

4.6 Smarter welcome mails

As described above, welcome mails are interrupting the onboarding process. However the possibility to store easy retrievable information on every computer the user has set up his email account is a possibility unique to (welcome) mails. A few changes could enrich the users benefit from welcome mail and decrease the interruption.

- Welcome Mails should be sent out after the user has finished his first explorative session in the MOOC portal, not directly after login.
- Platform welcome mails should be personalized, based on the previous usage. For example if a user has already signed up for a course during his first session there might be no need to tell him where he can find the course list. Instead related courses could be displayed or tips and tricks could be provided.
- A course welcome mail could include an iCal Attachment containing the course start event.

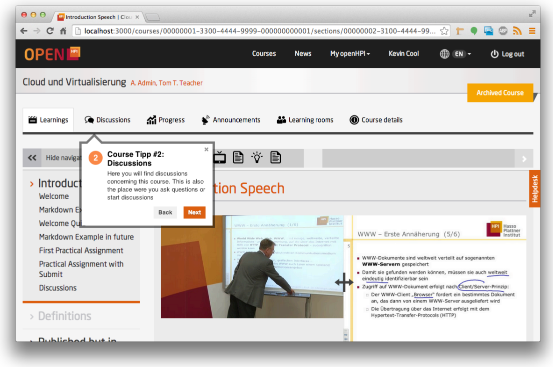
4.7 Segmented In-line Tutorials / Tours

As discussed in chapter 2, dedicated platform introduction courses or in-course platform descriptions have several drawbacks. One possible solution to this could be the usage of inline tutorials. Inline tutorials are integrated into the normal workflow. The platform itself should be aware of which tutorials should be shown when to a certain user, based on her current and previous actions.

In openHPI a dual-stream video player based on the tele-TASK[4] system is used. Therefore, a custom video player that is able to handle the synchronized playback of two video streams at is used. This video player offers some platform specific features, including automatically generated thumbnails, different playback speeds, and the possibility to resize the video streams. Those features are not common to other video players, so it can be assumed that many users will not be familiar with all or some of them.

[4] see https://tele-task.de

If a user visits a page containing this custom video player for the first time, the platform could present her a short introduction. There are some open source libraries like Hopscotch by linkedIn[5] or joyride by Zurb[6] that can handle this task and lower the implementation effort.



2 Course Tipp #2: Discussions

Here you will find discussions concerning this course. This is also the place were you ask questions or start discussions

Back Next

- WWW-Dokumente sind weltweit verteilt auf sogenannten **WWW-Servern** gespeichert
- Damit sie gefunden werden können, müssen sie auch weltweit eindeutig identifizierbar sein
- Zugriff auf WWW-Dokument erfolgt nach Client/Server-Prinzip:
  - Der WWW-Client „Browser" fordert ein bestimmtes Dokument an, das dann von einem WWW-Server ausgeliefert wird
  - Die Übertragung über das Internet erfolgt mit dem Hypertext-Transfer-Protocols (HTTP)

Figure 8: Prototype of an embedded tour explaining the course navigation

This guided tour consists of one or more overlays that are displayed next to a defined element within the User Interface (UI). The panel itself consists of a title, an instruction text, and a next and previous navigation. Furthermore, it can contain an optional call-to-action, which is basically a button than can direct the user to any URL. If the user clicks the next button, the next panel will be shown, if necessary, the page will automatically scroll to the position of the new panel. The new panel could also be positioned within a different page, so cross-page tours are possible. A user can easily close a tour by clicking the close button available in every step of the tour. Furthermore, if she ignores the tour the tour should automatically be closed after a certain time.

4.7.1 Fragmentation of Tours

The tutorial tours should be as compact as possible. Otherwise users wouldn't be able to jump to a certain step and skip a set of steps. This is due to the fact that they feature a linear navigation only. While they can be stopped and resumed later on, the authors of this paper suggest that this tour contains only explanations to one set of features.

In the example mentioned above a tour could look like this:

- Step 1**: Explore the openHPI video player**: Do you want to learn more about our new video player?
- Step 2**: Resize to your need**: Hover over the player and drag the slider to left or right to change the size of the videos.
- Step 3: **Customize the speed**: If the lecture is to fast or to slow, modify the play-speed here
- Step 4**: Know where you're going**: Use the segment navigation to easily jump to a certain slide

The steps are ordered by the probability that their corresponding features are unfamiliar to the user and the feature's relevance for course success.

4.7.2 Automatic detection of tour relevance

Users should only be presented relevant tours. This could be achieved by following these rules:

- Do not display a tour that a user has already taken or canceled
- Do not display a tour that explains features that the user has already used
- Remove those steps from a tour that describe features that the user already has explored

E.g. in the example given in section 4.7.1 step 4 should not be displayed if the user has already hovered over the navigation and saw the slide based navigation. To achieve this behavior the existing frameworks must be enriched by a smart event handling and a user based analytics data persistence.

[5] https://github.com/linkedin/hopscotch,

[6] http://foundation.zurb.com/docs/components/joyride.html

4.7.3 Passive vs. active tours

In the tour design it should be considered, which user interactions are required to proceed with the tour. Therefore, the authors of this paper would like to suggest a differentiation into passive and active tours. A passive tour consists of information about provided features. The navigation within the tour is independent from the fact if the users accessed this specific feature. In the example given above the user could easily walk through the complete tour without resizing the player, changing the playback speed or navigation.

In opposite to this an active tour would disable the “next” button and would require the user to perform a certain task to continue a tour. So, if the current step explains the resizing functionality, the user would have to actively resize the player. The tour then would detect that the user used this function and would trigger the tour to jump to the next step, ideally in conjunction with a positive feedback. Further more a mixed form could be offered, where the user can always navigate but some events triggered by user interactions will trigger a navigation of the tour as well.

4.8 Badges and Progress Bars

Badges and progress bars are two of the most used gamification elements in the e-Learning context. They could also be used to support the onboarding process. A progress bar displaying the current completion state of the users profile may motivate him to complete his profile by adding additional personal data. Badges could encourage the exploration of the platform. This will be discussed more in detail in an upcoming paper.

5 CONCLUSION AND FUTURE WORK

The paper has shown that onboarding a user is not as simple as easing the path to registration. Providing a good onboarding experience requires a deep understanding of the MOOC platform, the users needs and common web UI/UX patterns.

All six major MOOC providers that have been evaluated, are using at least some onboarding methods, however, none provides a consistent, sophisticated, orchestrated usage of onboarding techniques. Some suggestions for improving the onboarding experience have been introduced. Still an evaluation of these suggested improvements must take place after they have been introduced to openHPI or a comparable platform.

While these suggestions have been made in the context of MOOC platforms most of them are also applicable within traditional LMS.

Based on the more intense usage of MOOCs on mobile devices [7] further investigation has to be done how the onboarding patterns introduced in this paper are applicable within the mobile context and if specific mobile-only onboarding patterns can be identified.

REFERENCES

[1] http://en.wikipedia.org/wiki/Onboarding, last visited on 20.5.2013

[2] Zichermann, Gabe, and Christopher Cunningham. "Gamification by design." *Implementing Game Mechanics in Web and Mobile Apps* (2011).

[3] Deterding, Sebastian, et al. "Gamification. using game-design elements in non-gaming contexts." *PART 2-----------Proceedings of the 2011 annual conference extended abstracts on Human factors in computing systems*. ACM, 2011.

[4] Staubitz, T., Renz J.,Willems C., Meinel C., Supporting Social Interaction and Collaboration on an xMOOC Platform. In Proceedings EduLearn 2014 (Accepted) 2761/

[5] Willems, Christian, Renz, Jan, Staubitz, Thomas, & Meinel, Christoph. 2013. Reflections on Enrollment Numbers and Success Rates at the openHPI MOOC Platform. In: Proceedings of the European MOOC Stakeholder Summit 2014.

[6] http://developer.sonymobile.com/2014/04/29/gamify-the-ui-of-the-smartphone-with-the-evolutionui-research-project-open-source-video/, last visited on 20.5.2013

[7] Renz J., Staubitz T, Meinel C., MOOC 2 Go, in Proceedings of the 10th international conference on Mobile Learning 2014