

Sprint Retrospective Document

Date: 28/11/2018

Project acronym: COW

Members:

- İdil Zeynep Alemdar - e2098721@ceng.metu.edu.tr
- Sevim Seda Çokoğlu - e2008662@ceng.metu.edu.tr
- Ozan İncesulu - e2099711@ceng.metu.edu.tr
- Mohammed HAJI KARIM ALNAJAR - e2001535@ceng.metu.edu.tr

Supervisor: Dr. Onur Tolga Şehitoğlu.

Sprint 2 summary (Note: we have accidentally put the same ID for all Items):

Item ID	Workpac kage ID (from the Kick-off doc)	Status	Description
1	WP2	In Progress	<p>1. Create main actors and entities for the CMS (like students, instructors, courses)</p> <p>Description: Actors like student, instructor, course and conference are created as back-end components. This is necessary because they represent the main components of our project.</p> <p>Problems faced:</p> <ul style="list-style-type: none">- Nested entities were confusing at first. Courses have instructors and also students in them. Also, instructors have courses.- Functions related with every entity took some time because we had not decided on the permissions then.

			<p>Actions taken:</p> <ul style="list-style-type: none"> - Course, instructor, student, conference entities are created in Typescript by using mongoose. - Their functions and some example views are created as well. <p>Solutions found:</p> <ul style="list-style-type: none"> - Solutions for nested entities are found from the mongoose manual. - Functions are created temporarily and changed after the roles are decided. <p>Conclusion: The main actors for the project are created and their functions are implemented as well. The development of the roles will be done in the upcoming weeks.</p>
2	WP2	Complete	<p>2. Create bindings tool for the deployment orchestration using K8S</p> <p>Description: We are planning to use Kubernetes for providing networking, fail-safety, scaling and providing a vendor-independent service, however we don't want to expose our users to raw Kubernetes files, therefore we need to create a mapping tool that would create the cluster and allow deployment of different containers to the cluster</p> <p>Problems faced:</p> <ul style="list-style-type: none"> - Cluster initialization needed special attention since different databases needed different requirements - Default ingress controllers required many load balancers to be created which creates unnecessary complexity and cost <p>Actions taken:</p> <ul style="list-style-type: none"> - Do more research

			<p>Solutions found:</p> <ul style="list-style-type: none"> - Use NGINX Ingress Controller that creates a pre-configured NGINX server on top of one load balancer <p>Conclusion: We have worked in creating the basic tooling for Kubernetes.</p>
3	WP2	In Progress	<p>3. Do research and create simple user authentication and role management.</p> <p>Description: We need to create user authentication based on some fundamental roles</p> <p>Actions taken:</p> <ul style="list-style-type: none"> - A middleware has been created for authorization to be used by different components - Basic login-logout mechanism that uses a different endpoint has been created using simple login logout strategy or LDAP. <p>Conclusion: We need to create more fine-grained access that can contain more roles and entity-based authorization according to our advisor to simplify our workload for future.</p>
4	WP2	In Progress	<p>4. Create a course view with both public and private components.</p> <p>Description: A course page view is developed according to the requirements of COW that shall efficiently provide the functionalities of COW to their subjected users such as students, instructors and admins.</p> <p>Problems faced:</p> <ul style="list-style-type: none"> - The course templates are different from each other and there is a need to find a course page that would proficiently deliver the needed content to the user. - The content is huge which would cause a long course page that would result in a poor user satisfaction.

			<p>Actions taken:</p> <ul style="list-style-type: none"> - Basically, a research about the related problems was conducted, and various design options were examined some of best course management systems online. <p>Solutions found:</p> <ul style="list-style-type: none"> - After examining a variety of course pages from COW and oduclass we could finalize the course view. - We noticed that the way to show courses without the burden of scrolling over long pages was by creating clickable segments with declarative titles to lead the user into the needed section of the course page. <p>Conclusion:</p> <ul style="list-style-type: none"> - The course page was successfully completed and previewed by the supervisor. - The course page is very good for student view but for teachers it is more convenient to develop it in a manner that suits their needs. - The teacher's interface of the courses that they teach shall be studied and developed.
5	WP2	In Progress	<p>5. Design the CMS view with a good UX for better user adoption.</p> <p>Description: The course management system interface requires a front-end framework set up that is capable of supporting various pages needed by COW.</p> <p>Actions taken:</p> <ul style="list-style-type: none"> - After making a research about a suitable framework and setting it up we have started developing the view pages. - We started with a login page which also contain the publicly accessible elements of COW. - The student dashboard was developed which included notification capabilities.

			<p>Conclusion:</p> <ul style="list-style-type: none"> - COW front-end view looks modern and user friendly. - Building the course management system interface requires various pages including the built ones and more pages to be developed in the next sprint. - The user permissions and capabilities add an extra work on the interface part where some functionalities of COW require special view that are yet to be built.
6	WP3	In Progress	<p>6. Implement a service for syllabus construction from templates or existing pdf documents.</p> <p>Description: For syllabus construction tool, we need to make a general template. Template should be done via collecting different syllabi from other universities. The collected syllabi should be parsed as well to get different contexts.</p> <p>Problems faced:</p> <ul style="list-style-type: none"> - Syllabi are collected manually because pulling them via a code automatically was difficult since every website does not allow it. - Finding tools for parsing took a lot of time. <p>Actions taken:</p> <ul style="list-style-type: none"> - First, text files of the collected syllabi are extracted from pdf files with TET PDFlib. - It turns out that TET PDFlib did not include proper conversions for extractions. - The pdf files are converted to HTML files. - The lines containing 'Bold' are extracted. <p>Solutions found:</p> <ul style="list-style-type: none"> - Instead of using TET PDFlib, we used PDFMiner from python to convert pdf files to HTML files. - HTML files are parsed with a package from Python called BeautifulSoup but the version for python 3. <p>Conclusion: The parsing was done because we wanted to</p>

			see different syllabi all around the world and we will design the parsing of the syllabi according to these results. First step for the syllabus parsing was achieved.
7	WP4	Not started	<p>7. Create a global search service for different indexable entities.</p> <p>Problems: After revision the global search is considered by itself another WP and It is not part of WP2 anymore.</p> <p>Solutions: this task is moved into the related sprint.</p>

Sprint 3 plan

Item ID	Workpackage ID (from the Kick-off doc)	Description	Status
1	WP2	Create main actors and entities for the CMS (like students, instructors, courses)	Leftover from Sprint 2
3	WP2	Do research and create simple user authentication and role management.	Leftover from Sprint 2
4	WP2	Create a course view with both public and private components.	Leftover from Sprint 2
5	WP2	Design the CMS view with a good UX for better user adoption.	Leftover from Sprint 2
6	WP3	Implement a service for syllabus construction from templates or existing pdf documents.	Leftover from Sprint 2
7	WP4	Create a global search service for different indexable entities.	Leftover from Sprint 2
8	WP2	Create Front-end logic for smooth backend binding	New

Overall progress

	Sprint 1	Sprint 2	Sprint 3	Sprint 4	Sprint 5
MF1	5%	60%			
MF2	0%	30%			
MF3	10%	35%			
MF4	0%	0%			
MF5	5%	5%			
MF6	0%	0%			
MF7	10%	10%			
MF8	0%	10%			
MF9	20%	30%			
MF10	0%	0%			
MF11	0%	10%			
MF12	0%	0%			
MF13	0%	0%			
MF14	0%	0%			