SOLUTION DOCUMENT

Online Learning Platform

AUTHOR: ÅSA WEGELIUS, CLOVIS LEBRET, TUDOR STOICA
OWNER: ÅSA WEGELIUS
CLIENT: JARL TUXEN
VERSION: 1.0.5

1. Solution Document History

1.2 Revision History

version	Revision date	Implemented by	Reason
1.0.0	08-03-16	Åsa Wegelius	Ch. 5.6
1.0.1	09-03-16	Åsa Wegelius	Ch. 5.3 ,5.4 ,6, 7.2
1.0.2	09-03-16	Clovis Lebret	Ch. 5.8
1.0.3	10-03-16	Åsa Wegelius	Ch. 3, 3.2, 3.3, 3.4, 4, 5.2, 5.4
1.0.4	10-03-16	Åsa Wegelius	Ch. 5.5 from Ionut Vieru System Requirement
1.0.5	10-03-16	Tudor Stoica	Ch. 9 Risks

1.3 Approvals

Version	Name	Title	Date

1.4 Distribution

Version	Name	Title	Date
1.0.5	Jarl Tuxen	Steering Committee	11-03-16

1.5 Confidentiality Rating

Rating	
Company Confidential	x
Non Confidential	

1.6 Link to online version

Solution Document

2. Table of Contents

Table of Contents

3. General

The physical classroom is losing its monopoly as the only learning method, since the arrival of world wide web students can access information and learn from everywhere they are in the world just by having an internet connection and an online learning platform that help them learn long variety of subjects from economic to programming languages to philosophy and literature. With online learning platform students can learn and implement their learning on their own pace and time.

In this project we will develop the backbone of an online education platform. It will support three roles, Admin, Content Provider and Student. Admin administer the system, Content Provider produces and update courses and Student takes courses. The outcome of the project will be a prototype that fulfil this functions and can be extended to a further advanced learning platform.

3.2 Solution summary

About 2 billion out of 7 billion humans speaks English today. A platform restricted to English rules out 5 billion humans.

Around 40% of the population have an internet access today. This is just those that have their own personal internet connection. There are other ways to access internet, e.g. from a friend's home, from your job, from a cafe.

We will give all those that want to share their knowledge and all those that want to learn more a platform where they can connect. It will be designed to support an increasing amount of languages.

The platform will be an online site with access to variation of courses on subjects which are introduced by experts in those areas. After going through each part of the course the student can review and exam himself or herself on the learning by going through multiple choice tests.

People and enterprises interested in using the platform are paying a subscription per user which is going to be specified later on.

We will offer a platform that is

- Easy to access. All with a browser and a login can use it.
- For those with knowledge they want to share but are not English speaking.
- For those that want to study but don't know English.

3.3 Deliverables summary

- Browser interfaces for Students, Content Providers and Admin
- Web services for Students, Content Providers and Admin
- Databases for Users and Courses

3.4 Cost summary

Platform cost

The system will run on Wegelius private server at no cost. If or when the system will outgrow its current location is impossible to foresee and the future platform cost is therefore not feasible to tell.

License and support

Company	License/service	Amount/year
DNS Made Easy	DNS Failover	$29.95
		Total: $29.95

Operational Cost

Operational Expense	Amount/year
Weekly maintenance	2 man-hours/week = 104
Monthly maintenance	4 man-hours/month = 48
Quarterly maintenance	4 man-hours/quarter = 16
	Total: 168 man-hours/year

4. Recommendation and next steps

The idea with internet is that the more we share the smarter we get. This platform shares the same idea. All hardware is in place since we can share the setup SpeedVoter uses in the start-up phase. The only thing needed is the man-hour our team is prepared to deliver to make this idea come through.

5. Detailed solution description

5.2 Technical dictionary

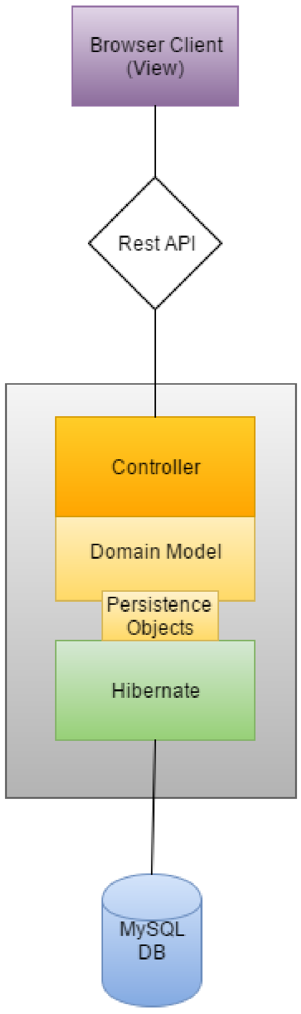
Term	Meaning
API	Application programming interface
DNS	Domain Name System, a hierarchical decentralized naming system for computers, services or any resources connected to the internet
Failover	A backup operation that automatically switches to a standby database, server or network if the primary system fails or is temporarily shut down for servicing.
Load Balancer	Distributes workloads across multiple computing resources, such as computers, a computer cluster, network links, central processing units or disk drives.
Persistence	What you save in a data storage to outlive the process that created it.
Rest API	An architectural style, and an approach to communications that is often used in the development of Web services
Sharding	A type of database partitioning that separates very large databases the into smaller, faster, more easily managed parts called data shards.
Web Service	An application provides a service to other applications and communicates it via a defined protocol over internet.

5.3 Architecture overview

I will look at the architecture from three different perspectives. The structure of the applications, the structure of the system and the structure of a Tomcat cluster.

The application structure

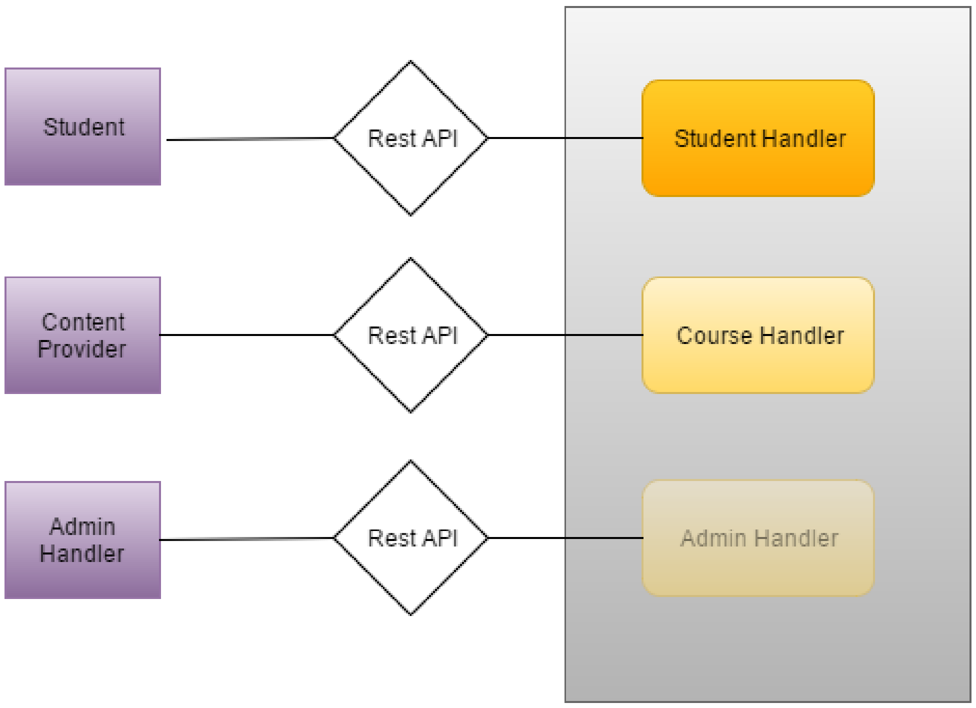
The frontend is a browser interface, the View. The Rest API is the communication interface between the frontend and the backend. The backend has a controller layer and a model layer. The persistence objects are a subset of the model layer. Hibernate is the ORM, that is, handles the object-relational-mapping between the application and the relational database (MySQL)



.

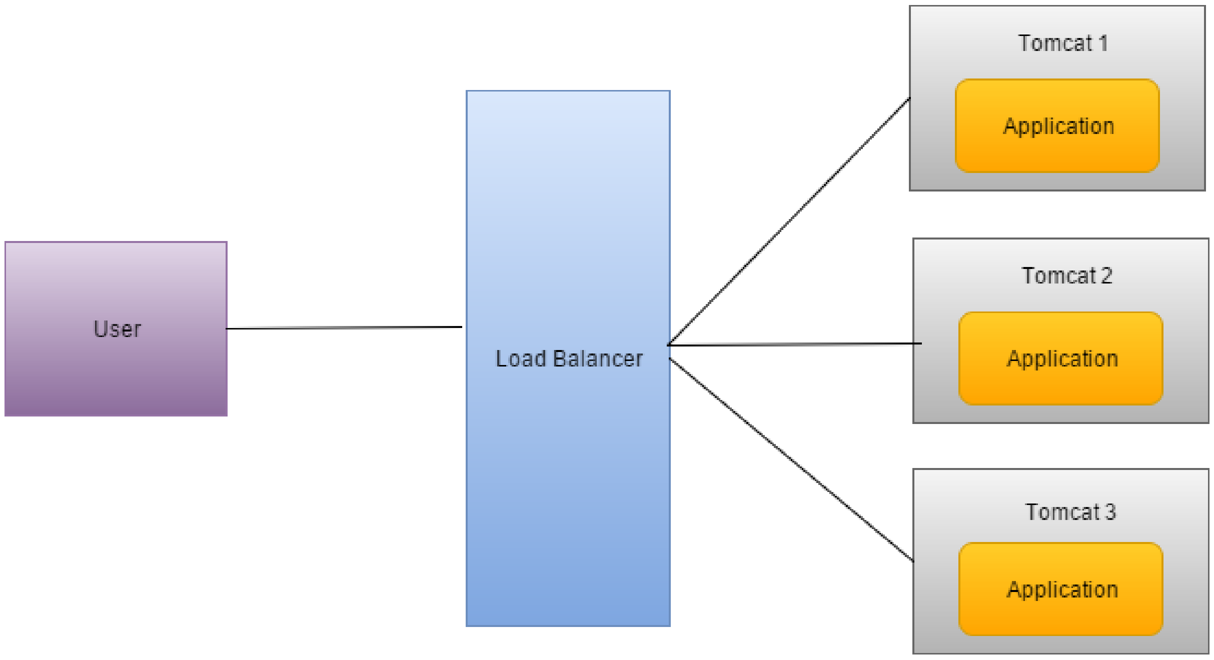
The system structure

The system consists of three separate restful services, one for Students, one for Content Providers and one for Administrators.



Tomcat cluster

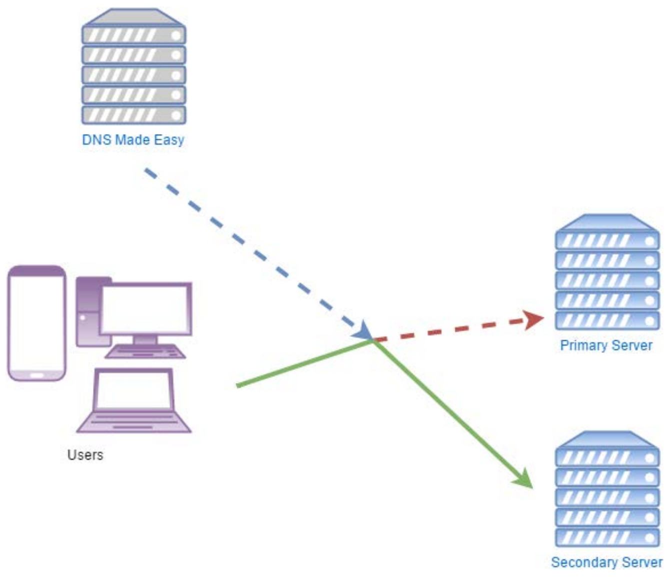
Tomcat have a recommended limit of 500 simultaneous users. This is the perspective of if there is a need to handle more.



5.4 Server setup

The system uses two databases, one with users and one with courses. The reason is that user information takes little space and can be stored at one location but courses contains videos which both require lots of space and as the product scales streaming uses an abnormal amount of bandwidth. So it will eventually require to be distributed, first to continent and then to countries.

The system is located at two servers, the primary server and the secondary (backup) server.



5.5 Functional requirements

Administrator - should be able to:

- login into the system
- register Content Providers
- create, edit, remove courses and description of the courses
- add/remove videos
- add, edit, delete exercises files
- create/edit/remove payment plans

Content Providers - should be able to:

- login into the system
- create, edit, remove courses and description of the courses they own
- add, edit, delete exercises files they own
- add/remove videos and courses they own

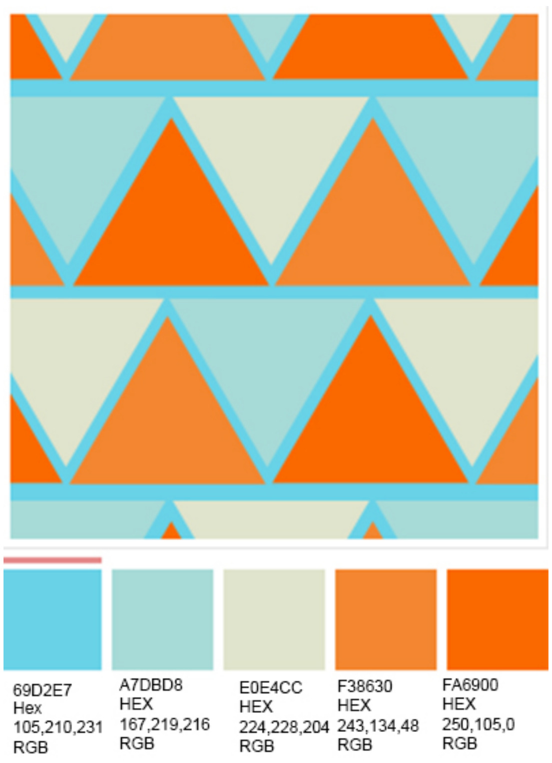
Regular users - should be able to:

- register into the system
- login into the system / logout
- edit profile (personal info, password, etc.)
- choose payment plans and make payments
- view courses and videos
- create playlists
- view progress of the courses, history, related or recommended videos
- submit done exercises and receive result
- search for specific courses

5.6 Non-functional requirements

Usability

Colour Palette for the User Interface:



69D2E7	A7DBD8	E0E4CC	F38630	FA6900
Hex	HEX	HEX	HEX	HEX
105,210,231	167,219,216	224,228,204	243,134,48	250,105,0
RGB	RGB	RGB	RGB	RGB

Training

A user shall not require any training to use the system. It shall be intuitive. A power user (Content Provider) need to know how to create videos

Accessibility

Users and Power Users shall need a browser and an internet connection to access the system

Localization

The system shall support English, Danish, Swedish and Arabic.

Reliability

Website Maintenance weekly check, downtime at the most 2h

- All pages on the front end of the website are loading without errors
- Check all forms to ensure they are working properly
- Review and resolve any issues with emails sent from the web server
- Check and remove spam comments, form submissions and user accounts
- Check for any broken links
- Check for 404 errors and resolve these by fixing links or redirecting

Website Maintenance monthly check, downtime at the most 6h

- Check website loading speed. If more than 3 seconds, then tweak/check the server
- Review security scans and resolve any issues

Website Maintenance quarterly check, downtime at the most 6h

- Test the website to ensure that it looks and displays properly on the most popular browsers and mobile devices
- Check backup health by restoring the most recent backup to a separate web server
- Check the uptime logs. If uptime is less than 99.9% then check/tweak the server

Supportability

Coding Standards

- Java Coding Standards
- HTML(5) & CSS Style Guide and Coding Conventions
- JavaScript Style Guide and Coding Conventions
- CSS and LESS Coding Standards

Design Constraints

Frontend

The web-based interface shall run on Explorer, Chrome, Safari.

Languages: JSP, HTML5, JavaScript, CSS, LESS

Backend

Java, Java Servlet API

Server

Apache Tomcat 8

Database

MySQL

5.7 Capacity recommendations

When manipulating scalable system, it's important to take in consideration the available capacity on the long term. Since we will be using a MySQL database as a storage solution, we will be provided options that limit the resources used by the backup process, in order to minimize backup overhead for busy or huge databases, or specify behaviours of the process if it has to encounter resource issues. On the future we might also imagine partitioning/sharding the database to additional server, making sure the system sustains.

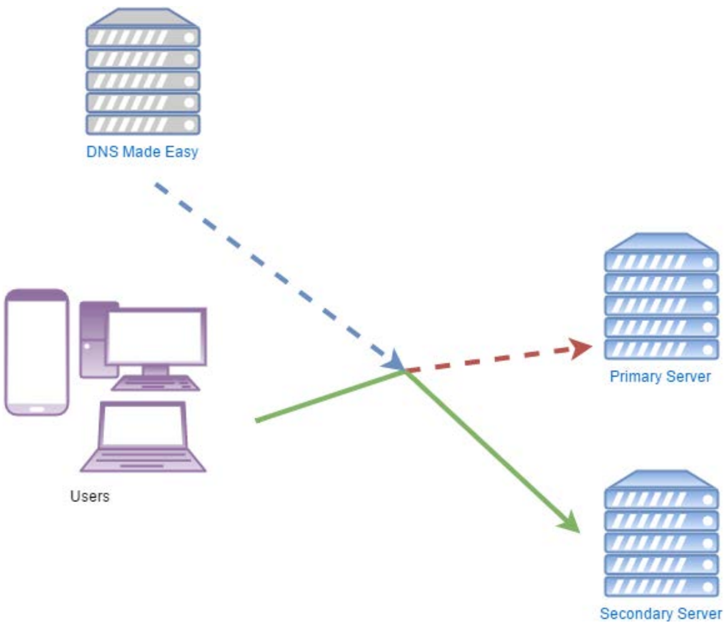
As for the capacity planning, it's very difficult to answer such a vast question of scalability. We might consider using MySQL Cluster, which is a highly scalable, real-time and provides automatic data partitioning with load balancing.

6. Impact on other system

SpeedVoter is installed on the same hardware server but runs in a different web server. SpeedVoter is an e-voting system with about 300 000 users. We will continuous monitor the performance to see that the server capacity can handle both.

7. Failover and scalability

The setup to handle failovers is use of a synchronized backup server. We will use a service called DNS Made Easy to monitor the server. It will check a script we setup and redirect to the secondary server if it fails.



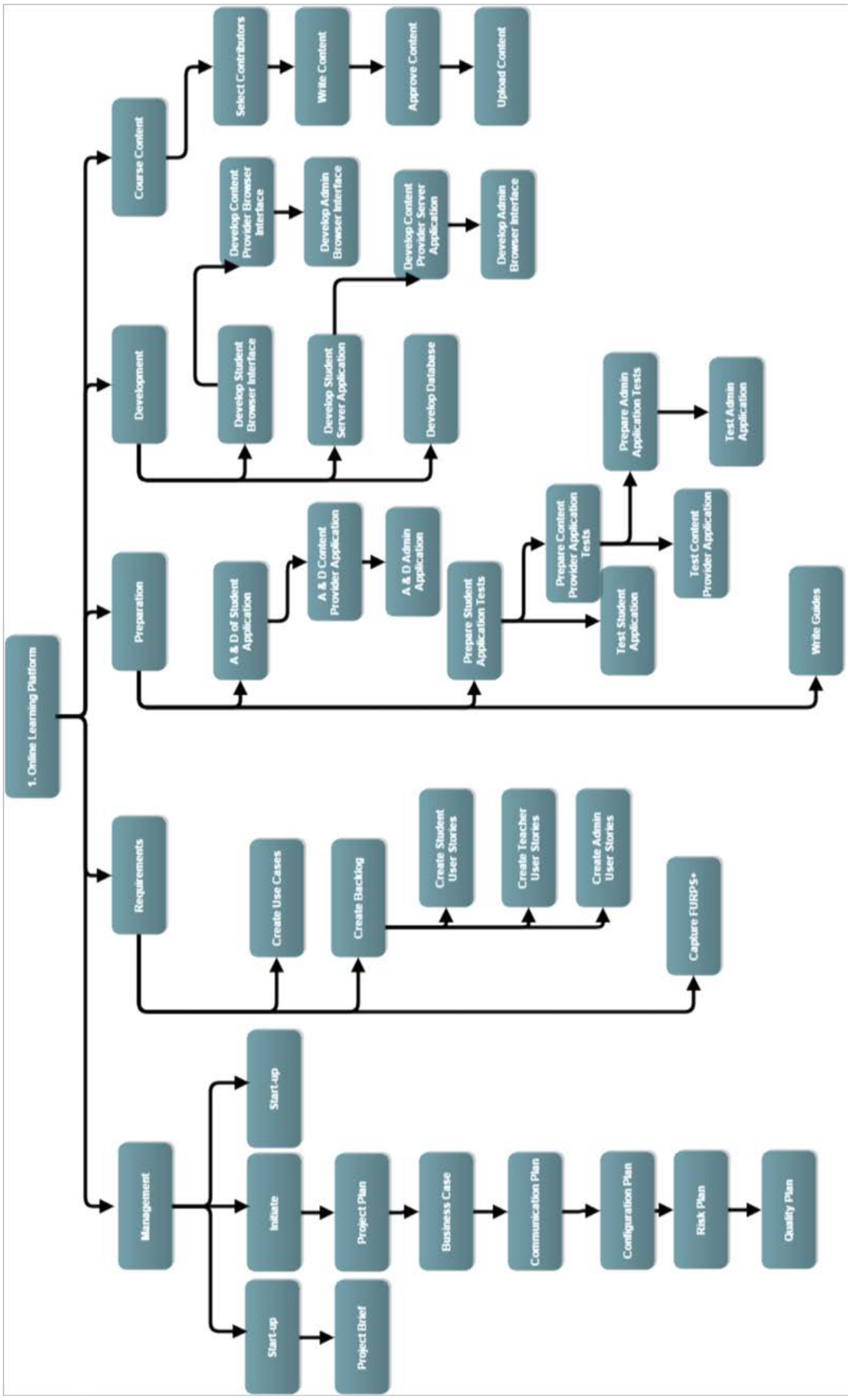
8. Technical implementation plan

8.2 Development Environment

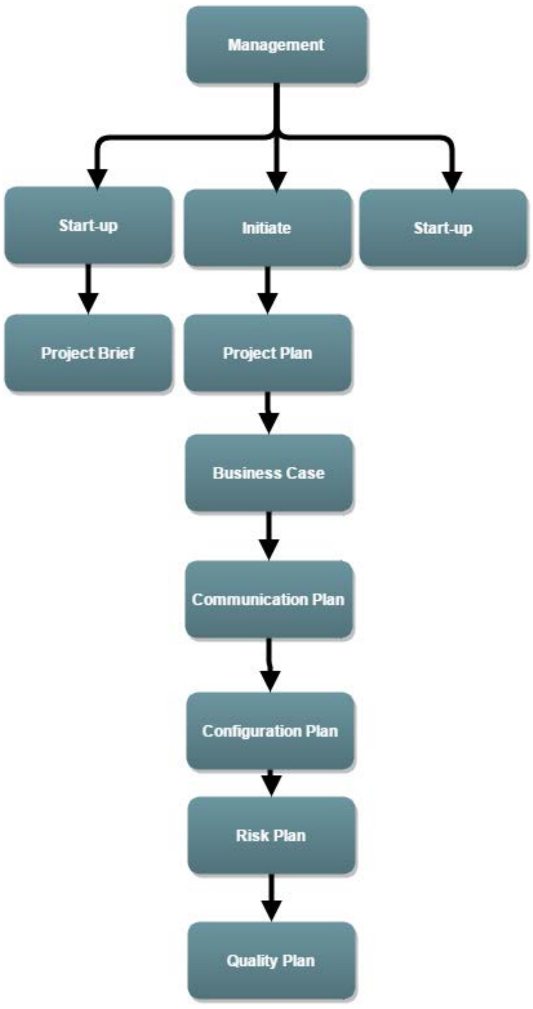
item	Applied for
Methods:	
Use Case	Requirement capturing
User Story	Requirement capturing
Supplementary Specification	Requirement capturing
Class diagram	Data modeling
Backlog	Requirement capturing
Tools:	

NetBeans	Construction
MySQL Workbench	Construction
GitHub	Version control
Draw.io	Modeling
Pencil Project	GUI modeling (mockups)
Languages:	
Java	backend
MySQL	DBMS
HTML5	frontend
JavaScript	frontend
CSS	frontend
LESS	frontend

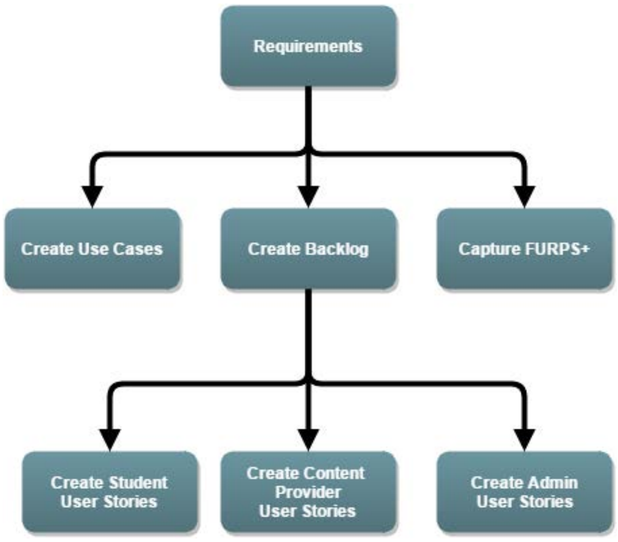
8.3 Solution implementation components (work breakdown structure)


1. Online Learning Platform
Management
Start-up
Project Brief
Initiate
Project Plan
Business Case
Communication Plan
Configuration Plan
Risk Plan
Quality Plan
Start-up
Requirements
Create Use Cases
Create Backlog
Create Student User Stories
Create Teacher User Stories
Create Admin User Stories
Capture FURPS+
Preparation
A & D of Student Application
A & D Content Provider Application
A & D Admin Application
Prepare Student Application Tests
Prepare Content Provider Application Tests
Test Student Application
Prepare Admin Application Tests
Test Content Provider Application
Test Admin Application
Write Guides
Development
Develop Student Browser Interface
Develop Student Server Application
Develop Database
Develop Content Provider Browser Interface
Develop Admin Browser Interface
Develop Content Provider Server Application
Develop Admin Browser Interface
Course Content
Select Contributors
Write Content
Approve Content
Upload Content

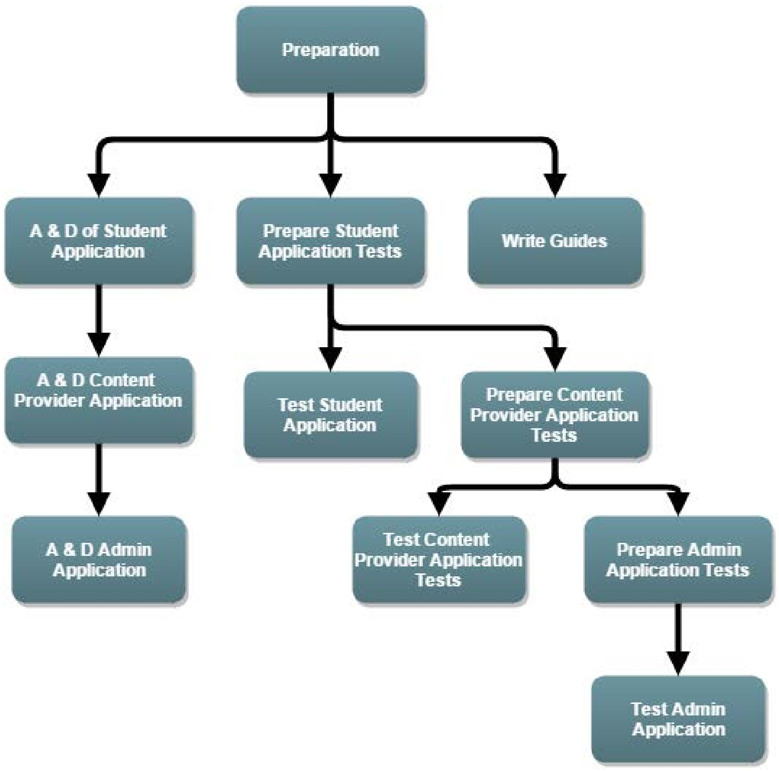
Management



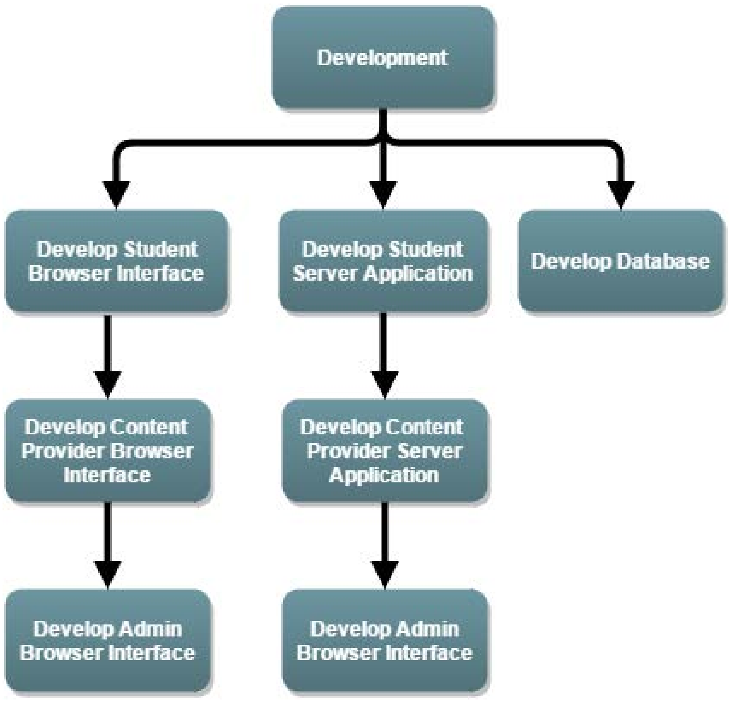
Requirements


Requirements
Create Use Cases
Create Backlog
Capture FURPS+
Create Student User Stories
Create Content Provider User Stories
Create Admin User Stories

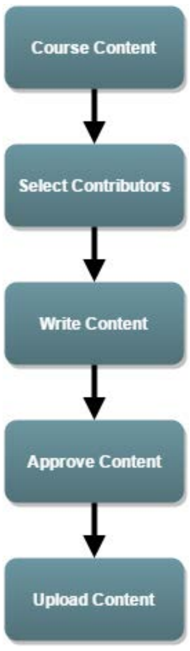
Preparation


Preparation
A & D of Student Application
Prepare Student Application Tests
Write Guides
A & D Content Provider Application
Test Student Application
Prepare Content Provider Application Tests
A & D Admin Application
Test Content Provider Application Tests
Prepare Admin Application Tests
Test Admin Application

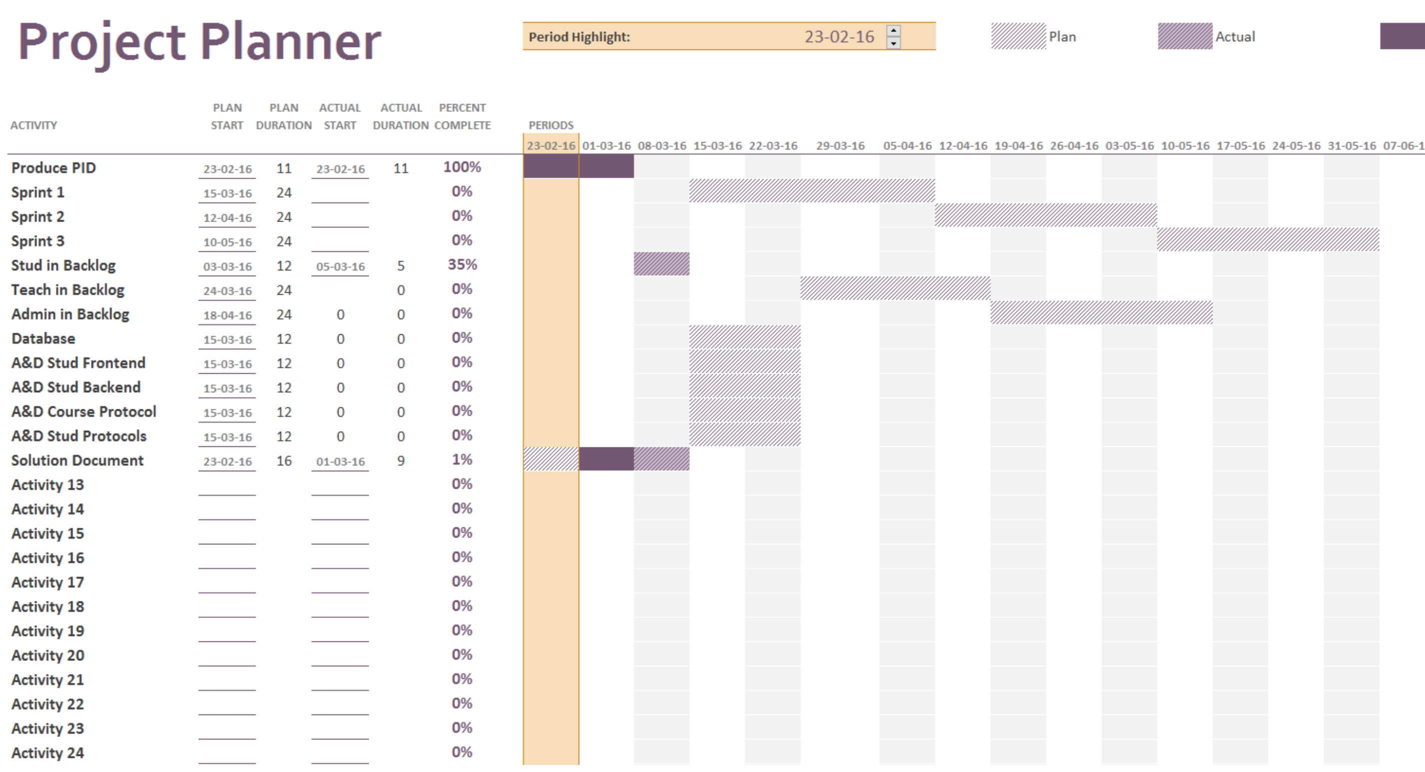
Development


Development
Develop Student Browser Interface
Develop Student Server Application
Develop Database
Develop Content Provider Browser Interface
Develop Content Provider Server Application
Develop Admin Browser Interface
Develop Admin Browser Interface

Course Content



8.4 Gantt Chart Plan



Project Planner

Period Highlight: 23-02-16

▨ Plan ▨ Actual

PERIODS: 23-02-16 01-03-16 08-03-16 15-03-16 22-03-16 29-03-16 05-04-16 12-04-16 19-04-16 26-04-16 03-05-16 10-05-16 17-05-16 24-05-16 31-05-16 07-06-1

ACTIVITY	PLAN START	PLAN DURATION	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE
Produce PID	23-02-16	11	23-02-16	11	100%
Sprint 1	15-03-16	24			0%
Sprint 2	12-04-16	24			0%
Sprint 3	10-05-16	24			0%
Stud in Backlog	03-03-16	12	05-03-16	5	35%
Teach in Backlog	24-03-16	24		0	0%
Admin in Backlog	18-04-16	24	0	0	0%
Database	15-03-16	12	0	0	0%
A&D Stud Frontend	15-03-16	12	0	0	0%
A&D Stud Backend	15-03-16	12	0	0	0%
A&D Course Protocol	15-03-16	12	0	0	0%
A&D Stud Protocols	15-03-16	12	0	0	0%
Solution Document	23-02-16	16	01-03-16	9	1%
Activity 13					0%
Activity 14					0%
Activity 15					0%
Activity 16					0%
Activity 17					0%
Activity 18					0%
Activity 19					0%
Activity 20					0%
Activity 21					0%
Activity 22					0%
Activity 23					0%
Activity 24					0%

9. Risks

Possible risks identified in the project's life cycle

Item #	Area	Description	Rank (RF=i*p)	Mitigation	Solution
1.	Performance	Lack of sufficient hardware to assure performance in real time	9=3*3	Evaluate the possibility to acquire preformat and scalable infrastructure	Migrate the application in a Cloud Infrastructure
2.	Security	Bad intended users can break the application with penetration tools.	6=3*2	Implement a security audit mechanism	Develop and implement security solution
3.	Development	Lack of time to develop the application	3=3*1	Evaluate the developing time allocated to each team member	Outsource the development process
4.	Software	Users might not agree with the graphical user interface	3=3*1	Prepare a UI survey	Ask an Web Design Specialist

Impact scale is from 1 to 3: the lowest impact of the risk is 1 and the highest impact is 3

Probability scale is from 1 to 3: the lowest probability the risk to occur is 1 and the highest probability the risk to occur is 3