[BDSA'21] Group Project Assignment

Version 0.5

Paolo Tell, Rasmus Lystrøm
Computer Science Department
IT University of Copenhagen, Denmark
pate@itu.dk - rnie@itu.dk

October 25, 2021

Version:

Version	Date	Note
0.1	13.oct.21	1st draft.
0.5	25.oct.22	Modified 2.6 and 2.3. Added Appendices A, B, and C.

1 Introduction

The group project associated with the "Analysis, Design, and Software Architecture" course (BDSA) is both instrumental for some mandatory activities of the course and an integral part of the final assessment. In particular, the project reviews and the project demo (see Table 1 for a quick overview of the activities) are mandatory activities of the course; and, the final course grade will be computed based on a 30/70 ratio between the project final hand-in and the written exam respectively. Note that both the project final hand-in and the written exam will have to receive a positive grade to successfully pass the exam.

Table 1: Overview of the project related activities.

Activity	Fridays	Mondays	Note
Group formation	Oct.15, 12:00 – 14:00		During exercise class
Project review 1	Oct.29	Nov.01	3
Project review 2	Nov.12	Nov.15	3
Project review 3	Nov.26	Nov.29	3
Project demo	Dec.10	Dec.10	4
Project submission	Dec.22, 14:00		Managed by SAP

The remainder of the document is organized as follows: Section 2 describes elements of the project activity in the context of BDSA; Section 3 and Section 4 detail the mandatory activities. Finally, Appendices A, B, and C contain descriptions of the project proposals available.

2 Formal Requirements

2.1 On the group

Groups must be communicated via email by Friday the 15th of October at 14:00. Each group must send one email to the course manager with the following information: group name, list of group members (including ITU user), preferred revision day (i.e., Mondays or Fridays). Revision days are only a suggestion; if necessary, groups will be assigned by the course manager to balance the load between the two days and among the TAs. Each group must comprise of 6 students[1]. Each group member must be actively involved in the projecy and participate to the mandatory activities. Each group will be assigned one TA that will follow the group through out the project. The teaching team will randomly join project activities.

During the exercise session of Friday the 15th of October between 12:00 and 14:00, you will have the chance to form groups. While the criteria for joining a group are many, we encourage you to remember considering: your experiece when working with some of your colleagues on the weekly mandatory activities, whether you share preferred work days with the team member, and whether you are interested in the same project topic.

2.2 Project portfolio

The project proposals have been inspired from the bank of projects proposed at the last SCORE competitions[2] organized under the international conference on software engineering (ICSE). In particular, three projects have been selected based on the content of the course, tweaked, and included. Hereafter, a summative description is provided.

Project bank. Providing awareness of project ideas on the one side and supervision needs on the other is a challenge that affects teachers and students every semester. The lack of a software technology able to provide a usefull yet intuitive and seamless platform to connect supervisors and interested students around a topic of interest cannot be suistained any longer.

The goal of this project is to create a matching service to connect student(s) with potential supervisors for projects and theses. The site will allow supervisors to post details of project ideas available and students to apply. This project will create a database-driven mobile-friendly service to encourage supervisors and students to engage in semester-long collaborations.

SE training. With the rapid evolution of software technology, including frameworks, testing and deployment tools, there is a need for a repository of high-quality software engineering resources. Students need access to tutorials and

[1] Special exemption must be discussed with the course manager

[2] http://score-contest.org

videos that they can use to self-learn. The resources should be better organised than just performing a search on Google. The resources should be lightly controlled to ensure inappropriate material does not appear, be easy to search, and allow for comments on the effectiveness of the learning materials.

The goal of the project is to build a repository of learning resources for software students to self-learn about important emerging technologies such as programming frameworks, testing tools, and deployment tools. Students need access to videos and tutorials that explain new tools and languages clearly. An example is a tutorial on React or a step-by-step account on installing Docker.

Lit Explore. Metaengines like GoogleScholar, Microsoft Academics, and the DBLP Computer Science Bibliography service provide bibliographic information on major (computer science) journals and proceedings. These services have become essential to academic communities. However, there is a lot of manual work involved in trying to perform any task more complex than performing a search.

The goal of the project is to provide (dynamic) features to allow users to explore the relations among papers. To exemplify the concept, a possible solution could visualize relations among papers as interactive graphs, in which each node represents a paper and directed edge between two nodes define references; a click on a node in the graph could show the pdf (if available), and a click on an edge could show the snippet of the paper citing another. Furthermore, the graph could be enhanced by users: marking edges as strong or weak to signify the importance of a relation or adding annotations to capture thoughts and comments are just some of the possible examples. These additional data could help other users to navigate through the graph more efficiently, and one could see the system also suggesing new relevant connection. It should be quite clear how this project project could be expanded in different directions once the domain is understood and the core is established.

2.3 Where are the details?

It is important to clarify that these descriptions are meant to guide the initial selection of the project. They provide a clear idea of the application domain for each of the projects as well as detail some high level directions that the project will take. Detailed descriptions have been added in Appendices A, B, and C. Plenary sessions will be organised when deemed necessary by the client. Dedicated sessions the client can be requested by teams on Wednesdays between 13:00 and 14:00 (CEST). Technical experts might be summoned if the necessity emerges from the meeting request.

2.4 On the project scope

The group is not only responsible for its formation and organization as well as for the selection of the project to work on, but also to find an original non-trivial vertical slice within the selected project to be delivered by the end of the course. All projects are of a size that can hardly be properly completed within the limited timeframe given, therefore, groups will have to define a vertical slice to focus on. The degree of originality of the idea will be considered in the grading of the final project. Note that a perfect non-original project with only a few minor weaknesses will still have access to a 12 grade. The originality of the vertical slice is not connected to the complexity of some algorithms present in the code base. The purpose of this project is to collaboratively design and develop a quality software product. The project should be sufficiently large to reasonably involve actively all group members. During the project presentation activity, you will be provided with feedback – if necessary – on whether we see issues regarding the project scope.

2.5 On the technical report

A strict requirement from BDSA is to execute the project following the sequential process described and used in the course book[3]. Groups are strongly encouraged to closely follow the document structures described in the course book, and, for such reason, Latex templates will be made available for the main documents to improve the likelihood that teams will focus on the content. You should keep in mind that the final document will have one table of contents including all the parts (i.e., RAD, SDD, ODD, . . .). On the one hand, the purpose is to provide you with a Latex skeleton that you can readily use – remember that using Latex is a must. On the other hand, the use of such template will avoid receiving documents with a font so small that a magnifying glass is necessary because "we need more space". The content of the technical report will be presented and explained throughout the course. As an initial indication, which might be subject of minor refinements in the future, the document will contain:

- the requirement analysis document (RAD)[4] (see Figure 4-16.OOSE);
- the system requirement document (SDD)[5] (see Figure 7-18.OOSE);
- the object design document (ODD)[6] (see Figure 9-15.OOSE);
- testing strategy (see below).

Note: the evaluators will stop reading after 50 pages; if you wish, you can put additional material as appendixes, but there is no guarantee that these will be assessed.

Details

Regarding the **RAD**, I only have one comment that I would like to clarify. The section regarding the scope is meant to describe the entire system that you are envisioning. Here you should explain what, according to the application domain,

[3] Object-Oriented Software Engineering by Bernd Bruegge and Allen H. Dutoit. 3rd. edition. 2010.

[4] Templates for the RAD will be provided.

[5] Templates for the SDD will be provided.

[6] Information for the ODD will be provided.

will be addressed by your system and what not. You might want to add to this section a subsection about the vertical slice. In this particular part, you would describe the exact de-scoping that you plan to do for the sake of the BDSA project. As an example, let's take Amazon. The application domain would be about selling goods to private individuals; the scope of the system would describe a platform that allows sellers to advertise their goods, and buyers to perform purchases; the vertical slide could limit the scope to the advertisement of the goods only (i.e., no purchasing).

Regarding the **SDD**, there is not much I would like to add, if not the fact that here you should be mainly focusing on your vertical slide. This might mean that some sections of the SDD become meaningless or less meaningful; this is ok, but you should still make us understand that you are providing no or limited content because it is not relevant/applicable. Access policies might be an example of this situation: you might end up with defining only access right for one specific subsystem accessible only by a limited number of actors, this is ok.

The **ODD** can be integrated in the SDD where relevant. For instance, you might detail design patterns and detailed class diagrams for specific subsystems directly in the SDD instead of the ODD. You might want to add an additional section to the SDD describing the class interfaces. Again, this is up to you. The book provides you a complete table of content, but it is ok to place relevant content in the SDD instead.

Regarding the **Testing**, it is important to have a document describing the strategy followed for the testing activities. There are several testing stages (i.e., from unit testing to acceptance testing), and you should describe your strategy for each of these. Additionally, after the strategy, you use this document to describe – where applicable – the results. For instance, with regards to unit testing, you might want to provide statistics about your coverage as well as passed, failed, and skipped information.

2.6 On the vertical slice

Given that there is no need to create a production ready system for BDSA, a useful guideline on how to scope the group effort is to identify a *vertical slice*. A vertical slice should illustrate a functional portion of a complete product. This could for instance be illustrated by one or more implemented use-cases.

Regardless of the specific decisions on it, the vertical slice must include connected components from:

- Front end (e.g. Blazor or Xamarin.Forms)
- API (ASP.NET Core Web Api)
- Services/repository
- Infrastructure (e.g. SQL Database or Azure Cosmos DB)
- Authentication/authorization

Additionally, the code must be delivered as a reference to a git repository. Finally, the code can be built and deployed as a set of containers running Linux.

3 Project Reviews

Where:	To be defined with the TA
Duration:	Roughly 20 minutes
When:	Depending on the group you are part of, one of the following slots.

	Fridays	Mondays
Review 1	Oct.29	Nov.01
Review 2	Nov.12	Nov.15
Review 3	Nov.26	Nov.29
	16:00 – 18:00	12:00 – 14:00

Purpose of the reviews and general guidelines:

- To have dedicated time with your assigned TA
- To make sure that your project progress is meaningful
- Do not expect the TA to be familiar with your project and/or artifacts
- All group members must be present
- Interaction of all team members is not necessary but encouraged
- The organization of the review is up to groups, the TA will stimulate conversations
- The time must be useful to you to discuss what you feel needs discussing
- Make sure that you get the TA to comment on the elements of the project that you wish to focused on
- Make sure that your TA is given access to the artifacts before the meeting

Below you can find some reflection points that will be used to guide the review sessions. Please do not consider this list by any means as comprehensive. I would expect the groups to identify discussion points that they want to be discussed as a regular occurrence during the meetings.

Focus points for review 1:

- Which project have you selected?
- What is the vertical slice that you will be working on?
- How are you planning to work together towards the target?

Focus points for review 2:

- Present the requirement analysis document (RAD) and system design document (SDD)
- Present the architecture

Focus points for review 3:

- What is the current status?

- Will you manage to complete?
- How is the project doing in terms of testing?
- Discuss non-functional requirements
- Demo the current status of the system

4 Project Demo

to be announced

A Project Bank

Project Description

Providing awareness of project ideas on the one side and supervision needs on the other is a challenge that affects teachers and students every semester. The lack of a software technology able to provide a usefull yet intuitive and seamless platform to connect supervisors and interested students around topics of interest cannot be suistained any longer.

The goal of this project is to create a matching service to connect student(s) with potential supervisors for projects and theses. The site will allow supervisors to post details of project ideas available and students to apply. This project will create a database-driven mobile-friendly service to encourage supervisors and students to engage in semester-long collaborations.

Project Scope

The system should support two different types of users: the supervisors, which can post new project and thesis ideas; and the students, which can apply for projects and theses as well as make unsolicited supervision requests. Both students and supervisors should be able to team up around a given project or thesis. This bare core can be extended in several directions, some of which are hereafter briefly listed.

- The service could allow users to create private or public threads around projects and theses with the goal of supporting a buzzing environemnt;
- The descriptions of projects and theses could be augmented either manually – by the creator – or automatically by including keywords describing the project. This would allow to include powerful searching features;
- The system should allow the inspection of classical statictics (e.g., the number of visits that a project or theses had in the last week/month).
- The service should allow users to subscribe to specific keywords to receive notifications once a project or thesis with the given keyword is uploaded or once students express their interest in specific topics.
- The service should allow users to describe themselves in terms of interests allowing other types of focused searches.

Environmental Constraints

Having the service deployed for mobile would be preferred, but the service must be provided through a Web interface at a minimum. The system can be developed in your language of choice.

Foreseen Extensions

A list of different directions have been listed in the project scope. Features beyond the list can certainly be discussed on a team basis.

Project License

The project should be run using the GPL FLOSS license.

Level of Sponsor Involvement

`#project-bank` on discord server. Teams are free to ping the client (i.e., the course manager) at any time, but answers may be delayed depending on the client availability. The client is happy to set up meetings with the teams to discuss project details on Wednesdays between 13:00 and 14:00 (CEST). The client will not respond to other forms of communication on the matter.

B SE Training

Note: Project taken from the SCORE 2021 Contest[7].

Project Description

With the rapid evolution of software technology, including frameworks, testing and deployment tools, there is a need for a repository of high-quality software engineering resources. Students need access to tutorials and videos that they can use to self-learn. The resources should be better organised than just performing a search on Google. The resources should be lightly controlled to ensure inappropriate material does not appear, be easy to search, and allow for comments on the effectiveness of the learning materials.

The goal of the project is to build a repository of learning resources for software students to self-learn about important emerging technologies such as programming frameworks, testing tools, and deployment tools. Students need access to videos and tutorials that explain new tools and languages clearly. An example is a tutorial on React or a step-by-step account on installing Docker.

The system should have a semi-automated function for populating materials, for example, automatically gathering the materials based on the provided links or sources. In addition, there needs to be a moderation function. There needs to be the ability to add a new capability. So for example, if there were no previous materials on Flutter, a new section of the repository should be easy to add. There should be some classification of the resources in the repository so that relevant resources are easy to find. How to implement search filters is up to the project team. The repository should allow two-way communication so that users of the resources can post (helpful) comments.

Project Scope

The resource repository should have a flexible search function, a moderator function for editing the material, and a capability that allows users to discuss the learning resources. Optional is a rating system for the quality of the materials. Clearly there are a lot of materials. Preference is for doing a more comprehensive repository on one category of technologies and tools, such as deployment, with the ability to extend, rather than a scattergun approach of all possible material. Teams should decide on which category is their initial focus.

Environmental Constraints

The repository needs to be accessible from a Web interface. So the system should be developed as a web-based application in your language of choice. People should be able to add resources from the Web interface, and link to the resources and/or download them.

[7] https://conf.researchr.org/home/icse-2021/score-2021. Accessed on October 20, 2021; original name: *TrainSE: Training resources for software engineering projects.*

Foreseen Extensions

It would be helpful if it were easy to create local copies of the repository.

Project License

The project should be run using the GPL FLOSS license.

Level of Sponsor Involvement

`#se-training` on discord server. Teams are free to ping the client (i.e., the course manager) at any time, but answers may be delayed depending on the client availability. The client is happy to set up meetings with the teams to discuss project details on Wednesdays between 13:00 and 14:00 (CEST). The client will not respond to other forms of communication on the matter.

C Lit Explore

Note: Project inspired from the SCORE 2018 Contest[8].

Project Description

Metaengines like GoogleScholar, Microsoft Academics, and the DBLP Computer Science Bibliography service provide bibliographic information on major (computer science) journals and proceedings. These services have become essential to academic communities. However, there is a lot of manual work involved in trying to perform any task more complex than performing a search.

The goal of the project is to provide (dynamic) features to allow users to explore the relations among papers. To exemplify the concept, a possible solution could visualize relations among papers as interactive graphs, in which each node represents a paper and directed edge between two nodes define references; a click on a node in the graph could show the pdf (if available), and a click on an edge could show the snippet of the paper citing another. Furthermore, the graph could be enhanced by users: marking edges as strong or weak to signify the importance of a relation or adding annotations to capture thoughts and comments are just some of the possible examples. These additional data could help other users to navigate through the graph more efficiently, and one could see the system also suggesing new relevant connection. It should be quite clear how this project project could be expanded in different directions once the domain is understood and the core is established.

Project Scope

A graph appears to be an adequate structure to model the data. Assuming this, and that publications are nodes, users should be allowed to create connections between the noted and annotate the edgse. This would allow users to create their own representation of links among publications of interest. The key to this project lies significantly on the ease of use and appeal of the user interface.

This bare core can be extended in several directions, some of which are hereafter briefly listed.

- Allow users to insert nodes from bibliographic files (e.g., bibtex files).
- Allow users to define different types of edges to represent different types of connections.
- Automatically detect the references used by a publications and create directed links between nodes accordingly.
- Allow users to dynamically explore the informations contained in the graph using a drill down methaphor through which inforamtion are presented to the users in more details based on the users' interactions.
- Allow users to search the graph.
- Allow users to create teams collaborating around a set of publications.

[8] http://score-contest.org/2018. Accessed on October 20, 2021.

Environmental Constraints

The repository needs to be accessible from a Web interface. So the system should be developed as a web-based application in your language of choice.

Foreseen Extensions

A list of different directions have been listed in the project scope. Features beyond the list can certainly be discussed on a team basis.

Project License

The project should be run using the GPL FLOSS license.

Level of Sponsor Involvement

`#lit-explore` on discord server. Teams are free to ping the client (i.e., the course manager) at any time, but answers may be delayed depending on the client availability. The client is happy to set up meetings with the teams to discuss project details on Wednesdays between 13:00 and 14:00 (CEST). The client will not respond to other forms of communication on the matter.