Tasks Affecting Grade

It is not required that all functionality of the product is implemented. Code what is needed to solve to the tasks in the list below, not more.

Grading

The tasks in the list below shall be solved by all group members together. Also, each group member shall, individually, write a report describing the solution to each task. The report must follow the provided template.

- Grade E
  - Accepted report describing the solution to all tasks marked X in the Mandatory column.
- Grade D
  - Accepted report, submitted no later than Mars 17, 2019, describing, and motivating, the solution to all tasks marked X in the Mandatory column.
- Grade C
  - Meet requirements for grade D
  - The report describes and motivates solutions to tasks with a total higher grade score of at least 7 (out of 20). You will only get score for tasks presented latest at the final group report, Mars 8.
- Grade B
  - Meet requirements for grade D
  - The report describes and motivates solutions to tasks with a total higher grade score of at least 12 (out of 20). You will only get score for tasks presented latest at the final group report, Mars 8.
- Grade A
  - Meet requirements for grade D
  - The report describes and motivates solutions to tasks with a total higher grade score of at least 16 (out of 20). You will only get score for tasks presented latest at the final group report, Mars 8.

Task	Mandatory	Higher Grade Score	Signature
Robustness			
Flexible, easily understood code, this *must* be motivated. It *must* also be motivated which patterns, code conventions and frameworks are used and how they are used.	X		
Appropriately layered architecture. *Motivate*!	X		
Javadoc comment for all public definitions.	X		
Security			
Authorization	X		
Authentication	X		
Logging, motivate *what* is logged, *how* it is logged and *where* the log is.		1	
Transactions			
Motivate *when* and *how* transactions begin and end.	X		
O/R Mapping and Database			
Motivate how primary keys are created.	X		
Management of relations, that is how relations in the database are translated into relations in the program. *Motivate* why the chosen solution is appropriate.	X		
Motivate how the content of the existing database is migrated to your new database.	X		
Error Handling			
Explain how the runtime errors (e.g. threw exceptions) are handled. Motivate why the chosen solution is appropriate. It should be clear where different errors are handled, if they are presented to the user (and if so, how) and whether and how they are logged.	X		

Task	Mandatory	Higher Grade Score	Signature
Internationalization and localization			
Internationalization and localization of web pages.		1	
Internationalization and localization of database.		2	
Process			
Justify the choice of functionality that has been implemented. It should be clear what requirements are solved by the written code. All existing code should solve a requirement from this list. *Write no code just to add more functionality.*	X		
Frequent integrations. Code developed by different group members should *often* be tested together and integrated in the group's shared repository.	X		
All code and documentation are stored *only* in a single repository of a version control system. This must work throughout the project.	X		
Use a build tool, for example Maven. Ant is not accepted.	X		
A continuous integration, CI, server is used for all development. Note that is must be used throughout the project, it is not enough just to get it started.		2	

Task	Mandatory	Higher Grade Score	Signature
Handover			
The application is live on a cloud platform, for example Heroku.	X		
The source code is available in a public Git repository, for handover to other developers, who did not take part in the development. You must write sufficient documentation to make it possible for those to continue development and deploy new versions.	X		
Testing			
Automated in-container unit testing, using for example *Spring TestContext Framework* **or** mocked-container unit testing, using for example the *org.springframework.mock.env* package. Choose one framework, you can not get points for both in-container and mocked-container tests. *The tests must be extensive, it is not enough just to get them going.*		in-container: 3 **or** mocked-container: 2	
Acceptance testing, using for example *Selenium*. The tests must assert that execution is correct, it is not sufficient just to record an interaction with the application. *The tests must be extensive, it is not enough just to get them going.*		1	

Task	Mandatory	Higher Grade Score	Signature
Reporting			
First report No later than February 8, 2019 (book reporting time in Canvas). You shall have code connecting all layers. An HTTP request shall result in execution of presentation layer, business logic and database. It shall be production quality code that can be left in the final product. You shall have a clear picture of what is done and what problems there are.	X	1 Given if the report is accepted before due date.	
Second report No later than February 22, 2019 (book reporting time in Canvas). You shall have substantially more code than at the first report (Feb 8, 2019). It shall be production quality code that can be left in the final product. You shall have a clear picture of what is done and what problems there are.	X	1 Given if the report is accepted before due date.	
Final report Performed no later than Mars 8, 2019 (book reporting time in Canvas). The group gives a detailed report to the supervisor.	X	1 Given if the report is accepted before due date.	
Individual written report Submitted in Canvas no later than Mars 17, 2019. The report must follow the template provided in Canvas. *The report is individual, you must not write it together with other students.*	X	2 Given if the report has high quality. This mainly concerns motivating your decisions, and providing references to documented best practices based on which decisions are taken.	

Task	Mandatory	Higher Grade Score	Signature
Other Functionality			
Validation of data entered in HTML forms. Validate in, at least, view and integration layers on the server.	X		
Views consist of different parts, e.g. header, footer, menu and main content.	X	1	
A REST server, which can be used by the mobile application, described in section 4.4 of the requirement application description. The REST interface must handle the use cases *Login* and *Show Applications*. You are not required to develop the mobile app, it is sufficient to test the REST interface using for example Postman, `https://www.getpostman.com/`		2	
Other Non-Functional Requirements			
Meet one or two non-functional requirement from the following list, explanations are found in the lecture notes from lecture one. It must be clear how you have proved that the requirement(s) are met. Meeting more than two requirements does not give any extra points. • Availability • Response time • Capacity • Scalability • Non-repudiation		1 point if one requirement is met 2 points if two requirements are met	
Your own choice			
You can define another requirement, not mentioned in this list. You must discuss the requirement with the teacher, and agree on how many higher grade points it is worth.			