ALPHA SOLUTIONS

Projektkrav

Generelle krav

Der skal være sporbarhed ml. de forskellige UP discipliner i jeres projekt (dvs. virksomhed, krav, design og kode og test). Det betyder, at I i rapporten skal sikre sporbarhed mellem jeres udarbejdede artefakteri faget Virksomheden til systemkrav og UML-diagrammer og videre fra UML-diagrammerne til jeres kildekode og test cases.

UML diagrammer og udsnit af kode skal ledsages af en supplerende tekst, hvor I beskriver jeres valg og reflekterer over fordele og ulemper ved de valg, som I har truffet.

Krav til Virksomheden

- Stakeholder analyse (Interessent analyse)
- Udvidet Risikoanalyse (Inklusiv Præventive tiltag og PlanB)

Krav til Systemudvikling

Systemudviklingen skal være baseret på Unified Process (UP). I skal arbejde iterativt, start derfor med at lave en plan for antal og varighed af iterationer i projektet. Husk start- og slutdatoer. Før hver iteration opdateres iterationsplanen, så det fremgår, hvilke use cases og øvrige artefakter I arbejder med i den kommende iteration. Iterationsplanen skal indgå i rapporten.

Følgende artifakter skal udarbejdes - Husk at det skal være læsbart for at være brugbart:

· Fuld Requirements list
· Liste over Functional/Non-Functionalor eller FURPS(+)
· Use Case diagram med alle identificerede Actors og Use Cases
· Vision, Glossary & Domain Model
· Minimum 3 Fully dressed Use Case
· Minimum 3 System Sequence Diagram af main success scenarie
· Liste med Operation Contracts
· Design Class Diagram - Et fra tidligt i processen og et som I endte med
· Package Diagram over jeres klasser
· State Machine Diagram – Et der illustrerer navigationsflowet i jeres applikation
· Minimum 3 Sequence Diagram for scenarier fra UC der skal, eller ønskes implementeret
· Beskrivelse af hvilke og hvordan i har anvedt GRASP i jeres program
· En tabel med testcases for de 2 implementerede use cases

Krav til Teknik

Jeres applikation skal Deployes til Heroku via Github og der skal være tilknyttet en database. Hvor denne database er hosted er op til jer, men i skal kunne begrunde jeres valg i rapporten.

Jeres projektsamarbejde skal forgå ved hjælp af git og github. I skal beskrive hvilket "Workflow" i har brugt i jeres projektsamarbejde (Centralized Workflow eller Integration-Manager Workflow) og forklare hvorfor i har valgt netop dettee.

I skal gøre brug af feature branches i jeres arbejde og i skal kunne forklare hvilke fordele der er ved denne tilgang frem for at lave alt i en master branch.

Det er et krav at alle i gruppen har bidraget med mindst en feature udviklet i en feature branch.

I skal redegøre for hvilke ændringer i eventuelt har gjort ved jeres applikation for at få den til at køre på Heroku. Herunder specielt de ændringer i har lavet i forbindelse med databasen.

Jeres github repository skal have en readmefil. I kan med fordel bruge den template for readme filen som vi har arbejdet med i timerne. Der skal i readme filen være en "step by step" beskrivelse af hvordan en vilkårlig bruger kan hoste jeres projekt på hans eller hendes egen Heroku konto.

Krav til softwarekonstruktion

Jeres løsning skal tage højde for følgende tekniske og designmæssige krav:

- Bruge en MySQL database.
- En logisk lagdelt arkitektur, der kan køre på en Java server (Tomcat).
- Applikationen skal bruge både almindelige Java klasser, Spring Controllere, HTML, CSS og Thymeleaf.

Produktkrav

- GUI er designet til brugerne med hensyntagen til Gestaltlovene og The Golden Rules.

Database

- SQL-script til oprettelse af database og testdata
- Et fysisk ER diagram, inkl. en beskrivelse af tabellerne i databasen (skema)
- En beskrivelse, analyse og vurdering af databasen, herunder til 3. normalform samt overvejelser om, hvorledes I har understøttet databasens integritet.

Programdokumentation

- Et implementeringsafsnit i rapporten med:
  - Beskrivelse af de programdele, der har særlig kompleks karakter (illustrér med indsættelse af kommenteret udklip af kode).
  - En beskrivelse og dokumentation af de løsninger, der kræver en særlig argumentation, f.eks. brug af design patterns, valg af datastrukturer, anvendelse af arv og/eller interfaces.
  - Kommentarer til programkoden om særlige forhold. Det kan f.eks. være:
    - Hvordan håndterer I exceptions?
    - Hvordan har I valgt at lave brugerinput validering?
    - Har I lavet sikkerhed i forbindelse med login – og hvordan?
  - SQL i løsningen (både DDL og DML).

Test

- I kan dokumentere automatiserede tests ved at beskrive i tabelform:
  - Hvilke klasser og metoder er testet
  - Dækningsgrad af jeres tests for de valgte metoder og klasser

Kørselsvejledning

- En beskrivelse af de softwaremæssige forudsætninger for at kunne anvende applikationen.

Krav til rapport og programdokumentation

I skal lave en projektrapport, der dokumenterer systemudviklingen og implementeringen af systemet. Rapportens målgruppe er underviserne.

VIGTIGT: I rapporten og i kildekoden skal det tydeligt fremgå hvem fra projektgruppen, der er ansvarlig for det pågældende kapitel, afsnit, diagram eller kodedel.

Projektrapporten udgør den skriftlige del af eksamen og skal ifølge studieordningen minimum indeholde:

- Forside med titel, navn og fødselsdato, hold betegnelse og dato
- Indholdsfortegnelse
- Problemformulering eller problemstilling
- Hovedafsnit
- Konklusion
- Litteraturliste (inkl. alle kilder, der er lavet henvisninger til i projektet)

- Bilag (inkluder kun bilag, som er centrale for rapporten)
- Der skal pagineres (sidetal) på alle sider
- Link til Github repository med kildekoden og github navn på alle deltagere.

Antal studerende	Max sideantal
1	40
2	50
3	55
4	60

En normalside er 2.400 tegn inkl. mellemrum og fodnoter. Forside, indholdsfortegnelse, litteraturliste samt bilag tæller ikke med i det afleverede antal sider. Bilag er uden for bedømmelse.
Hver enkelt figur eller diagram tæller 800 tegn.

Projektforløb og vejledning

Projektaflevering er den 17. december 2021 kl. 13.00 på Wiseflow.

Der gives vejledning i den angivne projektperiode. Vejledningen finder sted i forhold til det gældende skema og efter aftaler med vejlederne i de enkelte fag.