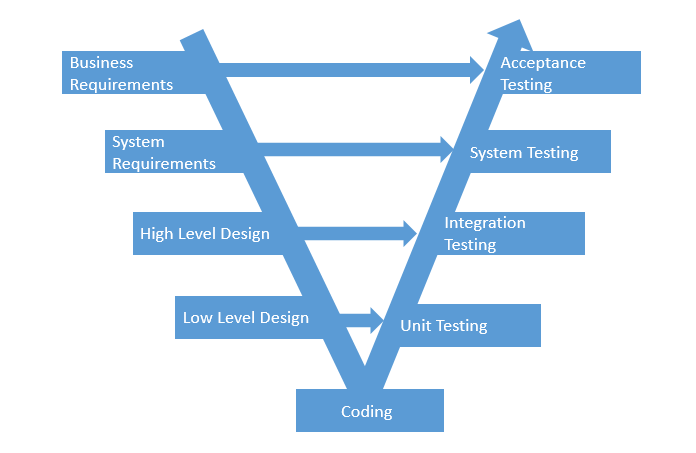
# 7 Test

|  |  |
| --- | --- |
| **Gruppenavn** | **666** |
|  |  |
| **Spørgsmål** | **Teststrategi** |
| Virker vores beregner? | Unit testing: White-box |
| Passer antallet af stolper? | Unit testing: White-box |
| Passer antallet af spær? | Unit testing: White-box |
| Andre beregninger? | Unit testing: White-box (Alle relevante metoder I beregner) |
| SVG-motor |  |
| Kan jeg tegne et rektangel med en vis størrelse? Virker vores metode? | Unit testing: White-box |
| Kan jeg tegne en pil som f.eks. er 300px lang? | Unit testing: White-box |
| Datamappere |  |
| Er der overhovedet hul igennem til databasen? | Integrationstest: Black box |
| Kan jeg indsætte en ny bruger? | N/A |
| Kan jeg hente et produkt / liste af produkter? | Integrationstest: Black box |
| Bliver en ordres status gemt i databasen når man opdaterer den på jsp siden? | Integrationstest: Black box |
| Hvad med vores brugergrænseflade? |  |
| Kan en kunde finde ud af at bestille en carport? Hvordan afgør vi det? | Testet på potentielle kunder |
| Kan de ansatte hos Fog finde ud af at bruge systemet? | Testet på ”Ikke-softwarekyndige”. |
| Hvordan håndterer vi input validering? | jQuery på front-end. Diverse validering på back-end. |
| Generelt |  |
| Er vores kode skrevet, så vi automatisk kan teste den? | Ingen selenium tests, men god struktur og base for unit- og integrationstests. |
| Hvordan sikrer vi os en ensartet kodekvalitet? | God dialog og daglig præsentation af gårsdagens arbejde. |
| Hvordan sikrer vi os at vi ikke tjekker fejlagtig kode ind på vores master- eller developerbranch i GitHub? | Ved altid at bruge *git pull* inden et commit og den daglige dialog. |



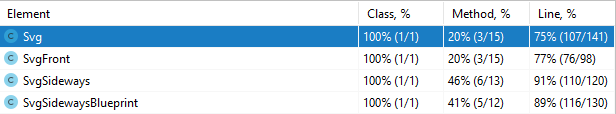
Ovenstående tabel beskriver måden hvorpå vi har testet vores system, samt hvilke niveauer vi dækker i forhold til V-modellen som ses til højre.

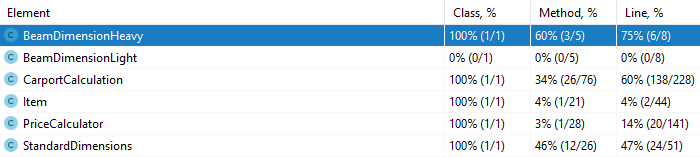
I hele vores projektforløb har der ikke været fokus på TDD grundet de kompetencer som vi har erhvervet os på 2. semester, ikke stemmer helt overens med TDD. Derimod har vi testet på klasser og metoder så snart vi har haft dem klar og næsten fuldt fungerende.

Vi har altså løbende i projektperioden testet og afprøvet tænkte scenarier hvor en kunde eller Fogmedarbejder kunne indtaste nogle forkerte værdier eller få programmet til at crashe. Derved har vi formået at fejlfinde på mange af de problemer et utestet system kunne forvolde. Derfor har test af systemets funktionalitet altid været en del af vores arbejdsgang og hvilket har sikret os en god struktur og et kompetent fundament for unit- og integrationstest.

I projektet er der både brugt white- og black box testning i form a Unit- og integrationstest, testet på ”ikke softwarekyndige individer” og jQuery på front end delen med diverse validering til back end. På den made har vi fået afhjulpet mange af de fejlen en bruger eventuelt måtte forsage allerede på front end delen.

Unit & integrations test resultater.







I ovenstående tabeller ses der vores dækning af de tests vi har lavet til projektet. Disse tests er nøje udvalgt grundet deres vigtighed og kompleksitet i opbygningen af projektet.

Hele svg motoren danner grundlag for alt det visuelle for bade kunden og for Fog medarbejderen og er derfor ret interessant at teste om giver det forventede udfald.Svg motoren er i vores projekt afhængig af CarportCalculations samt danner CarportCalculations også grundlag for den udregnede pris. Derfor valgte vi også at teste CarportCalculations og PriceCalculator.

Connecter og DataMapper er forbindelsen til vores database, som indeholder alle informationer om materialer og er derfor utrolig vigtig i forhold til PriceCalculator og CarportCalculations. Derfor har vi lave en række udvalgte integrationstest, på en testdatabase vi har sat op i WorkBench, som er vigtige for projektet.

*Der skal være lavet test. Du kan dokumentere tests ved at beskrive i tabel form:*

*Hvilke klasser er testet, Hvilke metoder der er testet, Dækningsgrad af dine tests for de valgte metoder og klasser*

*Desuden kan du beskrive hvordan i systematisk har arbejdet med at teste koden før den er blevet gjort til en del af master branch.*