

Betyg Betygskrav

- E Godkänt på samtliga labbar (inga bonusuppgifter) och projektet
- D Uppfyllt krav för betyg E samt redovisat och fått godkänt på minst 5 bonusuppgifter i tid. Alla labbar ska också vara klara på de ordinarie tillfällena.
- C Uppfyllt krav för betyg D samt:
 - Att man har designat ett REST API som en stor del av applikationen. En tjock klient kan vara att rekommendera här.
 - Specifikationen inlämnad och godkänd i tid.
- B Uppfyllt krav för betyg C samt ha byggt en app som i stor utsträckning använder sig websockets(eller likvärdig teknik) för att skapa interaktivitet mellan klienterna. Användningen av sockets ska vara ett naturligt designval och inte inklämt.
- A Uppfyllt krav för betyg B samt något av följande:
 - Implementera Native mobilklient för Android/iOS i stället för att göra en hemsida.
 - Eventuellt eget förslag som kräver betydande ytterligare kunskapsinhämtning. Denna kategori är huvudsakligen till för att inte hålla tillbaka överambitiösa studenter som är intresserade av nätverks-tung programmering.
 - Förslaget måste mailas till Vahid och godkännas!
 - Ex på godkända förslag: Torrent client, Torrent tracker.

Du ska göra en valfri uppgift baserat på ett etablerat ramverk för att generera en modern webbsida. Projektet är ett gruppprojekt på 1-3 personer.

Projektet redovisas vid separata redovisningspass som bokas via kurshemsidan i kursens slutskede.

Krav:

- Specifikation godkänd.
- Ramverket ska vara baserat på MVC eller likvärdigt komplex design. Projektet måste såklart vara implementerat enligt standarden för ramverket. Är du osäker, fråga.
- Visar god praktisk förståelse av DOM:en och hur man kan interagera med den (god användning av HTML, javascript och CSS). För A ska man visa motsvarande kunskap för design i mobilklienten.
- Har någon rimlig form av databasinteraktion.
- PER PERSON (minimumkrav för att vi ska kolla på projektet/förslaget)
 - MINST 500 rader kod, vill man vara säker på att vara godkänd bör man ha åtminstone 750 rader kod

- 2+ unika tabeller med 3+ kolumner per tabell givet BCNF. Bör helst ligga närmare 3 tabeller med 4+ kolumner om man vill vara säker.
- Rimlig säkerhet. Riktiga certifikat är för omständligt för den här kursen, men ni bör sätta upp den så att det hade varit säkert OM https användes. Ni förväntas inte implementera det själva utan bara kunna hitta färdiga implementationer som löser det åt er. Vanligaste saken är hashning av lösenord som skiljer sig mycket från annan kryptografisk hashning. Kolla på följande länk som referens: <https://gist.github.com/tqbf/be58d2d39690c3b366ad>
- Designen bör vara grundläggande Bootstrap eller bättre.

Kommentarer:

- Om ni är osäkra på något, fråga!
- Kursen är inte på något sätt en kurs i databasteknik men databasens storlek brukar vara ett relativt bra omfattningsmått på applikationen som arbetar mot den.
- Om ni justerar det betyg ni satsar på så är det bra om ni pratar med en assistent först då det kan påverka vad som krävs.
- Högre betyg -> högre krav. Om ni satsar på ett högre betyg så finns det högre krav på kodkvalitet, design, tekniker och omfattning utöver de explicita kraven. Ett A ska betyda något.
- Vi rekommenderar att man gör det i par. Är man ensam så blir ofta kunskapsinhämtningen för tung såvida du inte har tidigare erfarenhet inom att göra hemsidor. Är man tre så blir det mycket mer komplicerat att samarbeta och dela upp arbetet.
- **Du förväntas komma in för feedback!** Eftersom vi vill låta folk välja fritt hur de ska göra sitt projekt så har vi ganska lösa krav. Detta gör att det finns mycket utrymme för missförstånd. Detta har vi löst genom att ge generösa tidsramar för projektet och flera chanser att redovisa. Du bokar sista redovisningstillfället på egen risk.
- Bokningstiderna kan aldrig ta slut på sista dagen. Skulle de göra det ändå så ger vi dispans och sätter upp reservtider.

Specifikation

Innan ni påbörjar implementeringen ska ni skriva en specifikation av ert projekt. Denna utgörs av:

- En enkel funktionsbeskrivning på något tiotal rader.
- Ett databasdiagram med relationer.

Specifikationen godkännes av en handledare vid ett labtillfälle. Denna bör vara klar då ~2/3 av kursens labtillfällen genomförts, men senast på sista ordinarie tillfälle.

Exempelidér:

1 person D

Ett forum byggt i Laravel.

tabeller: trådar, inlägg, användare

Sidor: huvudsida (där trådar listas), trådsida (där trådens innehåll visas), användarsida (där användarens info visas).

kommentarer:

Går att utöka till 2 personer och B, man lägger till medelanden/chatt, man live-feedar nya kommentarer och ändringar av poster och man lägger till subforum samt admins.

2 personer B/A

Ett kösystem byggt med en backend i Express och med en frontend i Angular(B)/Android(A)

tabeller: Kö, person, kommentarer, log-entry, admins, messages.

Sidor: Kö-listning (där alla köer listas), kö-sidan (där själva kön visas), admin-sidan (där man kan lägga till och ta bort admins och göra inställningar för kön), statistik-sidan (där man kan göra enkla uppslagningar för att få ut statistik för kursen).

Kommentarer:

Om ni vill vara 3 personer eller vill ha inspiration be en assistent visa vilken funktionalitet som finns i det nuvarande systemet. Det mesta kan bara ses om man är assistent/lärare.

2 personer B/A

Ett spel byggt med en backend i Express och med en frontend i Angular(B)/Android(A)

tabeller: person, highscore, matcher, chatthistorik.

Sidor: Inloggning där man skriver in användarnamn och lösenord, användar-sida där man kan ändra sina personliga detaljer, highscore/historik där du kan se hur folk har gjort historiskt, lobby där du kan se de andra spelarna som vill spela, själva spelet.

Själva spelet kan vara hur enkelt som helst, ni kan ha att man ska skaka på mobilen (gyroskop), klicka på knappar när de dyker upp, ovs. Det viktiga här är att ni feedar datan till servern så att man kan se live hur det går. Tanken är att man ska spela mot varandra för att se att datan feedas ordentligt. Ni skulle annars kunna ha en annan monitor-sida om ni vill köra single-player.

Kommentarer:

Vill ni vara 3 personer så får ni verkligen hitta på något mer komplicerat på den här. Den är lite i tunnaste laget för två personer som den är och man borde hitta på något mer för ett A.