

# *OHJ-5100 Seittiohjelmointi*

Tentissä ei saa käyttää kirjallista materiaalia, laskimia tai muita lunttausvälineitä.

1. Vastaa seuraaviin kysymyksiin lyhyesti.

- a) Mikä on URL?
- b) Mikä on ORM?
- c) Mitä eroa on HTTP-protokollan GET- ja POST-metodeilla?
- d) Mikä on sivumootorin (template engine) tehtävä ja kuinka sellaista käytetään?
- e) Miten toimii HTTP-protokollan uudelleenohjaus ja mitä hyötyä tällaisesta on ohjelmoijalle?
- f) Mitä tarkoitetaan web-sovelluksen ohjelmitavalla rajapinnalla?

2. Autentikointi web-järjestelmissä.

- a) Mitä autentikoinnilla tarkoitetaan ja miten se eroaa autorisoinnista?
- b) Selitä, protokollatasokin läpikäyden, kuinka HTTP Basic -autentikointi toimii.
- c) Kuinka HTTP Basic eroaa HTTP Digest -autentikoinnista?

3. Kerro välimuistista (cache) web-järjestelmissä. Käsittele ainakin

- missä välimuisti voi sijaita
- miten välimuistin käyttöä voi ohjata, ja
- miten se vaikuttaa ohjelmien käyttäytymiseen.

4. Kerro Cross-Site Request Forgery (CSRF) -hyökkäyksestä sekä keinoista sen estämiseksi.

5. Toimit Django-konsulttina asiakkaalle, jonka web-palvelu on toteutettu käyttäen joukkoa Perl-ohjelmointikielellä kirjoitettuja CGI-BIN-skriptejä. Asiakas on käyttänyt moderneja web-sovelluksia ja haluaa uudistaa omia järjestelmiään. Tehtävänäsi on kertoa asiakkaalle teknisestä näkökulmasta mm.

- mitä hyötyä ja haittaa on käyttää Djangon tapaista web-sovelluskehystä
- miten MVC-malli toteutuu Djangossa
- kuinka toteutetaan asiakaspään ohjelmointi sekä kommunikointi palvelimen kanssa

Asiakkaalla on myös yhteistyökumppani, jonka web-sivuston sisältöä hän haluaisi sisällyttää omaan sivustoonsa. Esittele toteutusvaihtoehdot.