

Laboration 5 – Förberedelseuppgifter

1. Den abstrakta basklassen som alla gest-klasser ärver från heter UIGestureRecognizer - vilka konkreta subclasser finns det? Vilken typ av gest motsvarar respektive klass?
 - UILongPressGestureRecognizer
Användaren håller med en eller fler fingrar på en vy en kortare tid innan aktionen triggar igång
 - UIPanGestureRecognizer
Användaren håller ner en eller flera fingrar på en vy och drar över skärmen
 - UIPinchGestureRecognizer
Användaren använder två fingrar som den drar mot varandra på skärmen, används för att zooma ut oftast
 - UIRotationGestureRecognizer
Användaren använder två fingrar och flyttar dem i en cirkulär rörelse på skärmen
 - UISwipeGestureRecognizer
Användaren sveper med ett finger
 - UITapGestureRecognizer
Användaren gör ett eller flera tryck på skärmen
2. Hur använder du UISwipeGestureRecognizer för att hantera en "swipe" rörelse från höger till vänster? Gör egna antagande om variabler och dess typer (förklara dock vad som är vad).

Kod för att hantera en swipe rörelse från höger till vänster

```
UISwipeGestureRecognizer * swiperight=[[UISwipeGestureRecognizer  
alloc]initWithTarget:self action:@selector(swiperight:)];  
    swiperight.direction=UISwipeGestureRecognizerDirectionRight;  
    [self.view addGestureRecognizer:swiperight];  
//Metod för swipe-right  
-(void)swiperight:(UISwipeGestureRecognizer*)gestureRecognizer  
{  
    //Vad du vill att den ska utföra  
}
```

3. Om du inte skulle använda klassen ovan för att hantera swipe-rörelsen, hur skulle du då bära dig åt? Vilka metoder behöver du implementera? Du behöver inte skriva någon kod eller förklara i yttersta detalj, huvudsaken är att du visar att du förstår skillnaden mellan dessa metoder. Vilken av dessa två sätt att hantera gester (alltså att manuellt hantera gester, som du beskrivit här eller att använda UIGestureRecognizer-klasser) tycker du verkar enklast/bäst?

Det skulle behövas implenteras en egen bounce animation samt en swipe interruption ifall användaren swipear åt andra hållet. Alltså är det mycket bättre att använda UISwipeGestureRecognizer som redan har allt implementerat, ser ingen anledning till att manuellt försöka implementera en swipe-rörelse

4. Med en MKMapView vill man ofta visa användarens position. Hur går man till väga för att göra det på ett enkelt och användarvänligt sätt?

Koden nedan lägger till en knapp i navigationsmenyn, vilket automatiskt zoomar in på ens position i mapview om man trycker på den.

```
self.navigationItem.leftBarButtonItem =  
[[MKUserTrackingBarButtonItem alloc] initWithMapView:  
self.mapView];
```

5. För att lägga in kartnålar (annotations) i en kartvy behöver flera klasser samverka. Förklara hur det går till när en kartnål visas, vilka sorts klasser som behövs och vad de har för uppgift.

Först behövs det en UITapGestureRecognizer som registrerar ett tryck på kartan sedan måste den kopplas till en metod som skapar en kartnål när användaren tryckt på skärmen. Sedan måste metoden skapa en CGPoint för att veta var på kartan användaren tryckt, sedan används CLLocationCoordinate2d för att konvertera trycket på skärmen till kartkoordinater. Sen skapas en MKPointAnnotation som kopplas till CLLocationCoordinate2d för att skapa en pin på de kartkoordinaterna.