

Jag har valt att planera en sjukvårds applikation med den agila metoden. Jag föreställer mig att jag har ett företag som kund, som har anlitat mig för att skapa deras första applikation. De har alltså inte haft någon tidigare version av plattformen.

Planeringsfas

I början av projektet så gör jag en övergripande planerings fas. Detta är för att i stora drag få en klar bild av vad som behöver utvecklas så att man kan planera sina sprints.

Forskning och behovsanalys: Man intervjuar kunden och för att få en förståelse över vad deras idé av appen faktiskt går ut på. I just denna appen säger vi att kunden vill ha: en applikation där patienter kan boka en online tid med en läkare. Det ska finnas både videomöte och chattmöte. När man söker så ska man skriva vad ens problem är, och sedan få en tid. Man ska också kunna se ens journal.

Nu när vi vet vad kunden är ute efter så kan man börja tänka på vad ett sådant system behöver rent praktiskt. Jag kom fram till:

1. Användarprofil och inloggning:
2. Tidsbokningssystem
3. Kommunikationssystem, video och chatt
4. Journalhantering
5. Databas
6. Säkerhet och dataskydd
7. Information/påminnelse-system
8. Betalningssystem

Nu när vi i stora drag har planerat vad som behöver göras så delar vi upp arbetet i 2 veckors sprinter vars ordning prioriteras baserat på kundvärde och komplexitet. Funktioner med högst kundvärde och minst komplexitet ska göras först. Varje sprint följer de fem systemutvecklingsstegen. Det sista steget att lansera och underhålla ser lite annorlunda ut jämfört med vattenfall. Man lanserar inte till världen efter varje sprint, utan visar applikationen till intressenterna för feedback på programmet, eller "lanserar" till en mindre grupp människor för feedback. Planen är inte heller exakt eftersom det kan hända att man behöver gå tillbaka och göra om/ändra något efter intressenternas feedback.

Plan

Sprint 1: Tidsbokningssystem och databas

Tidsbokning är en central del av applikationen och har därför högt kundvärde, samtidigt som det inte är väldigt komplicerat att göra. Databas är nödvändigt för att tidsbokningen

ska fungera och är inte heller särskilt komplicerat eftersom vi inte ännu fokuserar på säkerhetsaspekten

Uppgifter:

- . Skapa en databas (vilket behövs för att kunna visa lediga tider och spara den bokade tiden)
- . Implementera tidsbokningssystem som kollar i databasen efter lediga tider och bokar sparar den nya tiden till databasen
- . Implementera avbokningssystem som tar bort tid och gör den tillgänglig igen för andra användare
- . Implementera användargränssnitt för tidsbokningen
- . Implementera att man kan bifoga en bild som läkaren kan se på förhand

Sprint 2: Kommunikationssystem och Journal

Tidsbokningssystemet och kommunikationssystemet utgör de två viktigaste funktionerna för användarna. Kommunikationssystemet har alltså högt kundvärde. Journalen skulle jag säga är den tredje viktigaste när det kommer till kundvärde, och kommer säkerligen att vara den mest använda funktionerna förutom att boka tid och kommunicera med läkare. Det är alltså också högt kundvärde. Det är också ganska lätt att tillämpa.

Uppgifter:

- . implementera videomötesystem
- . implementera chattsystem
- . implementera ett journalsystem där patients historik sparas för patienten och läkare att se
- . implementera användargränssnitt för video och chattmöte, samt för journal.

Sprint 3: Användarprofil och inloggningssystem, samt information och påminnelse system

Användarprofil och inloggning skulle jag säga är hyfsat högt kundvärde eftersom det är viktigt att ha ett smidigt inloggningssystem för att användarna ska vara nöjda, vilket intressenterna säkerligen är intresserade av att se hur det implementeras i deras app. Påminnelse system skulle jag säga är ganska lågt kundvärde men också ganska lätt att implementera.

Uppgifter:

- . Implementera system för att skapa konto
- . Implementera användargränssnitt för inloggning och skapande av konto
- . Implementera kontosystem där användaren kan skapa ett konto som sparas i databasen med ett användar-id som används för att koppla sparad information som tex journalen eller bokad tid till användaren.
- . Implementera rollhantering: olika konton för läkare och patient eftersom läkaren till exempel ska kunna se flera patienters journaler medan patienten bara kan se sin egen.
- . Implementera inloggning med bank id.
- . Implementera påminnelse-system: Programmet ska spara användarens kontaktinformation och önskade sätt att bli kontaktad: sms eller mail, i databasen. Sedan ska systemet automatiskt skicka ut en påminnelse tex en timme innan besöket.

Sprint 4: Säkerhet och Dataskydd, samt betalningssystem

Säkerhet och dataskydd är väldigt viktigt för en app med så känslig information och har högt kundvärde, men också hög komplexitet. Betalningssystem är av medel högt kundvärde eftersom det är viktigt att ha ett smidigt betalningssystem, men kanske inte lika intressant för intressenterna som att se tidsbokning, chattsystem eller dataskydd. Det är dock ganska lätt att implementera.

Uppgifter:

- . Implementera kryptering av användarinformation och kommunikationssystemet
- . Implementera behörighetskontroll för användarinformation och samtal
- . Implementera säker inloggning med bank id
- . Implementera betalning med olika betalningssätt som swish eller kort betalning.